



Hochschule für Angewandte Wissenschaften Hamburg
Hamburg University of Applied Sciences

Bericht INF-M3 Ringvorlesung

Markus Dreyer

Your Home in Your Hand: Get the specs

Inhaltsverzeichnis

Inhaltsverzeichnis	i
Abbildungsverzeichnis	ii
Tabellenverzeichnis	ii
1 Einleitung	1
1.1 Motivation	1
1.2 Szenario	1
2 Einordnung in existierende Lösungen	3
2.1 Heimautomatisierung	3
2.2 Heimserver	3
2.3 Bewertung	3
3 Anforderungen	4
3.1 Anforderungsanalyse	4
3.2 Fokussierung	5
4 Ziele	6
4.1 Abstraktion	6
4.2 Mobiler Klient	6
4.3 Verwendbare Standards	7
5 Risiken	9
6 Konklusion	11
6.1 Kooperation	11
6.2 Einsatzmöglichkeiten	11
A Literaturverzeichnis	a
B Glossar	b

Abbildungsverzeichnis

1	Szenario	2
---	--------------------	---

Tabellenverzeichnis

1	Komponenten des „Your Home in Your Hand“ Szenario	4
2	Priorisierung der Komponenten	5

1 Einleitung

Zur Integration von Geräten und Diensten in der heimischen Umgebung gibt es unterschiedliche Ansätze. Im Rahmen des Masterstudienganges Informatik an der Hochschule für Angewandte Wissenschaften Hamburg (HAW) soll eine Idee entwickelt und vorgestellt werden, die zur Aufgabe hat, eine personalisierbare und konfigurierbare Umgebung zu schaffen. Diese soll es einem Nutzer erlauben, Dienste zu nutzen oder in eine vorhandene Umgebung zu integrieren.

1.1 Motivation

Die hier entwickelte Idee soll als Grundlage für die Masterarbeit dienen. Die in Dreyer (2008b) vorgestellten Konzepte können ebenfalls als Grundlagen genutzt werden. Die Idee eine Wohnung von einer Zentrale aus verwalten zu können, ist nicht neu. Dennoch gibt es weder ein schlüssiges Konzept zur Integration unterschiedlichster Komponenten, noch eine Personalisierung der angebotenen Dienste.

Meine Freundin möchte ihre Fernsehaufnahmen grundsätzlich nicht über Showview programmieren, da sie befürchtet den Anfang oder das Ende würde dann abgeschnitten. Ich würde diesen Komfort normalerweise nutzen. Warum kann man diese Unterscheidung nicht in den Geräten hinterlegen, so dass ich nur die Showview Nummer eingeben brauche, während meine Freundin gleich Datum, Uhrzeit und Programm eingeben kann. Aus dieser Überlegung und der Tatsache der Fernbedienungsflut in meinem Haushalt, soll hier Abhilfe geschaffen werden.

1.2 Szenario

Das Szenario bewegt sich in einer häuslichen Umgebung und zeigt die Interaktionsmöglichkeiten und die Automatisierung von Alltagsabläufen auf. Eine detailliertere Beschreibung des Szenarios findet sich in Dreyer (2008b). Die Handlungen innerhalb des Szenarios spielen sich in der in Abbildung 1 gezeigten Umgebung ab. Es handelt sich um eine Wohnung die mit unterschiedlichen Geräten ausgestattet ist. Es wird davon ausgegangen, dass diese miteinander kommunizieren können. Für den Bewohner der Wohnung stellt sich nun die Frage nach einer einfachen und ubiquitären Möglichkeit mit den Geräten zu interagieren. Dabei soll die Steuerung sowohl als augenblickliche Interaktion, sowie mustergesteuert erfolgen können. Das Szenario beschreibt eine Person, die auf dem Heimweg nach der Arbeit einige Musikstücke kauft. Zu hause angekommen wird die gekaufte Ware direkt auf den PC heruntergeladen. Parallel dazu werden die E-Mails abgerufen und direkt auf dem Fernseher angezeigt. Während sich die Person später für die Zubereitung ihres Abendessens in die Küche begibt, bekommt sie einen Anruf, der automatisch über die Lautsprecher abgewickelt

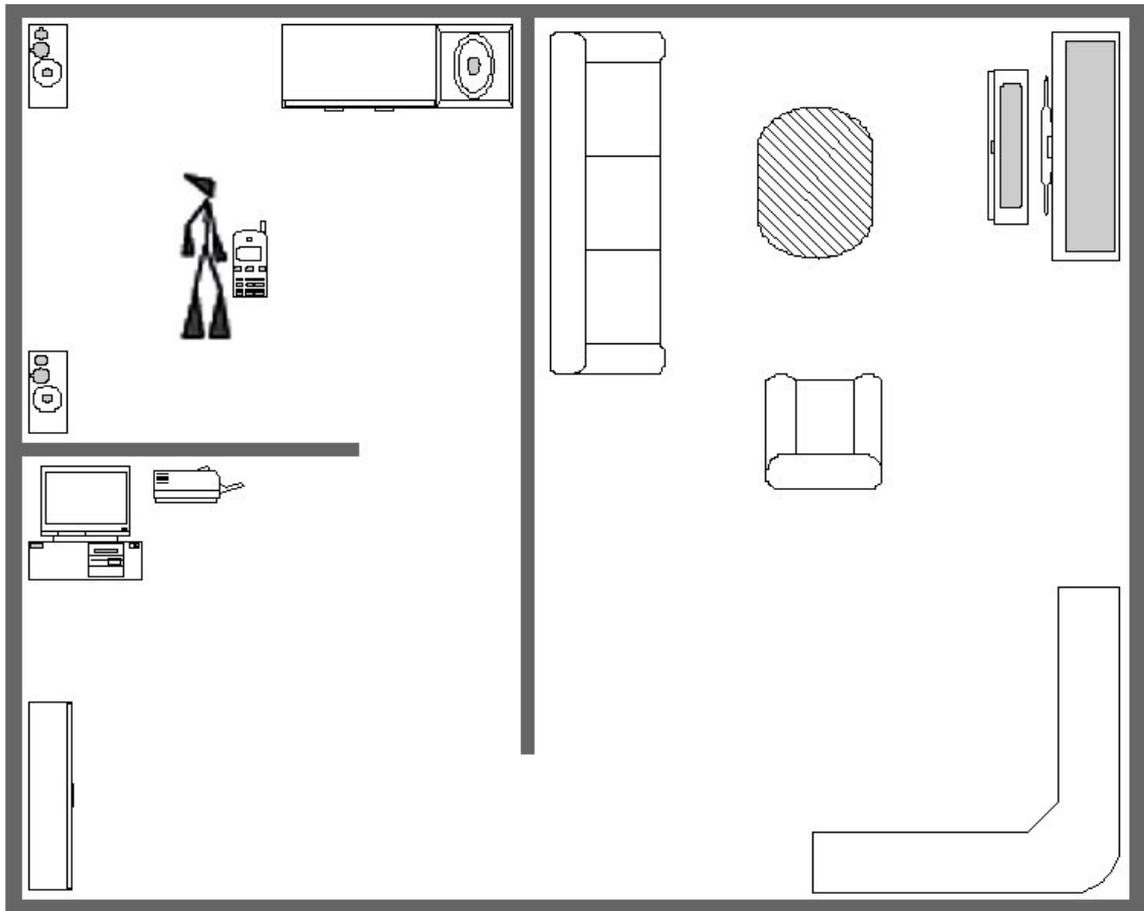


Abbildung 1: Szenario

wird. Da das Gespräch länger dauert, beauftragt der PDA automatisch den Festplattenrekorder mit der Aufzeichnung der Lieblingssendung. Nachdem das Telefonat beendet ist und der Bewohner aufgegessen hat, wird es Zeit für die Geburtstagsfeier eines Freundes. Der PDA hat zu diesem Zweck schon mal den Wetterbericht ausgewertet und schlägt warme Kleidung und einen Schal für den Abend vor.

2 Einordnung in existierende Lösungen

Schaut man sich das Szenario an, fallen Parallelen zu zwei Bereichen auf: Zum einen die Heimautomatisierung mittels einzelner Komponenten und zum anderen die zentrale Verwaltung vorhandener Dienste mit einem Server. Beide Bereiche werden im Folgenden näher erläutert. Vergleiche mit konkreten Implementierungen oder Ansätzen finden sich in Dreyer (2008b).

2.1 Heimautomatisierung

Im Bereich der Heimautomatisierung gibt es diverse kleine bis große Anbieter, die individuelle Lösungen erstellen. Diese legen sich meist auf ein System fest, welches die Erweiterbarkeit bestimmt. Als Standard in diesem Sektor ist X10 anerkannt. Dieser erlaubt vornehmlich das Schalten und Steuern von Geräten. Dazu zählen hauptsächlich Licht, Verdunkelung, Heizung und sonstige Großgeräte wie Herde oder Kühlschränke. Für Hifi und Video gibt es andere Standards, die nicht der Automatisierung, sondern viel mehr der Wiedergabe mittels unterschiedlicher Geräte dienen, sprich der Medienverteilung.

2.2 Heimserver

Ein Server, der alle erreichbaren Geräte steuern kann, bietet sich als zentrale Instanz an. Mit etwas Aufwand kann man den Heimserver, der momentan als Medienzentrale angeboten wird, zur vollständigen Steuerung der Wohnung umfunktionieren. Dieser stellt damit die Benutzerschnittstelle bereit und wandelt die Eingaben für die Geräte um. Somit lassen sich quasi beliebige Dienste integrieren.

2.3 Bewertung

Das größte Problem beider Lösungen spiegelt die Integration von unterschiedlichen Standards wider. Beide Lösungen sind für ihre Kernbereiche (Regelungen-Medienverteilung) gut geeignet und haben dafür eigene Standards kreiert. Sie lassen sich nur schwer auf beliebige Dienste erweitern. Weiterhin besitzt die Heimautomatisierung in der Regel keine Benutzerverwaltung (Zugriffsschutz nur rudimentär), während ein Server ohne zusätzlichen Komponenten Regelungssysteme nur schwer steuern kann. Die angestrebte Lösung bietet den Funktionsumfang beider Systeme, erweitert um eine Personalisierung der Dienste.

3 Anforderungen

In diesem Kapitel sollen die Anforderungen an die mobile Steuerung unterschiedlicher Geräte untersucht werden. Hierbei soll bereits ein Fokus auf die zu entwickelnde Anwendung gelegt werden. Um die Anforderungen zu erarbeiten, wurde in Kapitel 1.2 ein Szenario vorgestellt, welches gleichzeitig einen Ausblick für das Projekt darstellt.

3.1 Anforderungsanalyse

Die Anforderungsanalyse dient als Basis für die Bewertung vorhandener Ansätze und ist die Grundlage für das weitere Vorgehen zur Gestaltung der Teilkomponenten, die für eine Umsetzung des Szenarios benötigt werden. Aus dem Szenario ergeben sich unterschiedliche Teilbereiche die in Tabelle 1 aufgeführt und kurz beschrieben sind. Die hier definierten

Komponente	Beschreibung	Beispiel
Profilspeicherung	Anpassung der Dienste und deren Darstellung erfolgt benutzerabhängig.	Die Lautsprecher dienen der Wiedergabe des Telefonates nur dem Wohnungseigentümer.
Dienstverwaltung	Anwendung zur Steuerung des Zugriffs und der Nutzungsrechte von Diensten	Der PC darf Daten an den Fernseher übermitteln.
Dienstfindung	Verfahren zur Auffindung von Diensten z.B. Johanson u. a. (2004) oder Webservice	Der PDA beauftragt den Festplattenrekorder mit der Aufnahme.
Kontextabhängigkeit	Die Umgebung hat Einfluß auf die Dienstnutzung	Die Lautsprecher sollen nur genutzt werden, wenn der Hauseigentümer allein ist.
Authentisierungsverfahren	Nur berechtigte Nutzer sind zur Verwendung bestimmter Dienste zugelassen.	Der Download der Musikstücke ist nur dem Käufer gestattet.

Tabelle 1: Komponenten des „Your Home in Your Hand“ Szenario

Komponenten sind nur ein kleiner Teil der benötigten Infrastruktur zur Verwirklichung des Szenario. Da die Arbeit auf die Realisierung einer Anwendung zur Steuerung und Verwaltung der Dienste in einer bestimmten Umgebung abzielt, sind weitere Anforderungen zu stellen. Eine der wichtigsten ist die Definition einer Schnittstelle, über die die einzelnen Geräte miteinander kommunizieren können. Hierbei empfiehlt es sich - wie in Tabelle 1 bereits angedeutet - auf bestehende Standards oder Systeme zurück zu greifen. Auf eine semantische Lösung zur Dienstfindung soll vorerst verzichtet werden. Zur Integration des Kontextes

in die Anwendungen sollen nach Möglichkeit die gleichen Schnittstellen verwendet werden. Für die Authentisierung eines Nutzers oder eines Gerätes hat sich in den letzten Jahren Kerberos als quasi Standard entwickelt. Eine Integration dieses ticketbasierten Verfahrens in die Anwendung ist demnach wünschenswert, aber nicht Hauptbestandteil dieser Arbeit. Die Schwerpunkte der Arbeit beschreibt das folgende Kapitel.

3.2 Fokussierung

Das eingangs beschriebene Szenario ergibt eine Vielzahl von Entwicklungs- und Forschungsmöglichkeiten, die für eine Ausarbeitung im gegebenen Zeitrahmen zu mächtig ist. Aus diesem Grund wird nur ein Teilbereich von dieser Ausarbeitung bearbeitet. Hierbei handelt es sich um den Realisierungsansatz für eine Dienstverwaltung auf mobilen Endgeräten (PDA, Smartphone), da sich diese laut Koskela und Väänänen-Vainio-Mattila (2004) für die Verwendung im intelligenten zu Hause am besten eignen. Aufgrund dieser Annahme ergeben sich implizit Anforderungen an eine energiesparsame Anwendung und eine simple Benutzerschnittstelle. Aus den Erfahrungen des Masterprojektes - Dreyer (2008a) - wird deutlich, dass die Zukunft der Anwendungsentwicklung für Kleingeräte auf Java liegt (siehe Kapitel 4.2). Aus diesem Grund empfiehlt sich die Entwicklung der Anwendung mit J2ME oder Android als Grundlage. Für das kommende Projekt ist Android die beste Wahl, da hier schon ein funktionsfähiger Emulator zur Verfügung steht und die Geräte mit Android Plattform für kommendes Jahr angekündigt sind. Durch diesen Schritt kann die Anwendung auf einen breiteren Schatz an Bibliotheken zurückgreifen und somit von dem Umstieg profitieren. Weiterhin gibt es Bestrebungen, das Spieleframework des Projektes auf Android zu portieren, wodurch die Abstraktion der Hardware nutzbar ist. In das Framework ist weiterhin eine Webserviceschnittstelle integriert, die somit die Dienstnutzung ermöglicht. Für die einzelnen Dienste muss eine einheitliche Schnittstelle zur Steuerung und Verwaltung geschaffen werden die sich als Komponente in das Framework integriert. Mit dieser soll es möglich sein die aufgefundenen Dienste zu konfigurieren und an die eigenen Bedürfnisse anzupassen.

Die für die Masterarbeit relevanten Anforderungen sind in der folgenden Tabelle aufgeführt und mit einer Gewichtung versehen.

Weiterhin soll die zu entwickelnde Software den aktuellen Anforderungen der Softwareentwicklung genügen und Schnittstellen zur Erweiterung bieten. Ebenso muss berücksichtigt werden, dass die Anwendung auf mobilen Geräten laufen soll. Die dadurch entstehenden Einschränkungen in Bezug auf die vorhandenen Ressourcen und den Energiebedarf, sollen nicht vernachlässigt werden.

einfache Benutzerschnittstelle	+++
Sicherheit und Zugriffsteuerung	+++
Personalisierbarkeit	++
Konfiguration der Dienste	++
Dienstfindung	+
erweiterte Zugriffssteuerung	+

Tabelle 2: Priorisierung der Komponenten

4 Ziele

In diesem Kapitel werden die Ziele konkretisiert und mögliche Techniken zur Realisierung aufgezeigt. Dabei sollen nicht alle Techniken aufgeführt werden, sondern die aussichtsreichsten zur erfolgreichen Umsetzung.

4.1 Abstraktion

Ein wichtiger Punkt für die Umsetzung ist die Abstraktion der Dienste. Damit möglichst alle Geräte mit der Anwendung zusammenarbeiten, muss von unterschiedlich intelligenten Systemen ausgegangen werden. Ein Temperatursensor, der periodisch die aktuelle Temperatur versendet, hat andere Konfigurations- und Authorisierungsmöglichkeiten als ein DVD-Rekorder. Beide Geräte sollen ihre Dienste zur Verfügung stellen. Damit sind zwei gänzlich unterschiedliche Gerätetypen charakterisiert. Der Temperatursensor arbeitet nach einem Pushverfahren und stellt nur Daten bereit. Die Konfigurationsmöglichkeiten beschränken sich auf ein Minimum. Zur Authorisierung müssen die Anforderungen an einen weiteren Dienst geleitet werden, der die Zugriffssteuerung abwickelt. Der DVD-Rekorder arbeitet auf Anfrage eines Klienten und kann Benutzer dank seiner Speicher- und Rechnerkapazität selber verwalten. Für die Klienten soll der Zugriff transparent erfolgen. Also muss die Anwendung die konkreten Verhalten der Geräte verstecken. Gleiches gilt für die Dienstkonfiguration. Die Personalisierung erfolgt direkt auf den Klienten. Für diese Funktionalität soll automatisch ein Template mit den entsprechenden Daten gefüllt werden. Dieses ist durch den Nutzer anpassbar und erlaubt somit eine individuelle Gestaltung der Oberflächen. Weiterhin soll es möglich sein, ausgewählte Eigenschaften auszublenden und die Anzeigereihenfolge zu verändern. Dafür werden auf dem Klienten alle Anzeigeelemente verwaltet und mit entsprechenden Attributen versehen. Bei der ersten Kommunikation sollen die Elemente automatisch dargestellt und ein Template angelegt werden. Über die Software hat der Anwender danach Zugriff auf die Anzeigeeinstellungen. Geräte sollen für den Erstkontakt Standard-Templates mit übergeben können. Weiterhin wird an dieser Stelle eine Versionskontrolle benötigt. Damit die Anwendung auf Geräte, die keine Versionsinformation bereitstellen, reagieren kann, muss sie regelmäßig die Anzeigeelemente auf Änderungen prüfen.

4.2 Mobiler Klient

Die Anwendung kann mit unterschiedlichen Technologien umgesetzt werden. Die bekanntesten Entwicklungsumgebungen für Mobiltelefone sind J2ME und .NET. Im letzten Jahr wurde das Google Phone - Android- vorgestellt. Diese drei Plattformen stehen zur Auswahl für die Umsetzung und sollen kurz vorgestellt und bewertet werden.

Die J2ME Plattform ist am Markt etabliert und seit der Spezifikation unterschiedlicher JSR (Java Specification Request) standardisiert. Die Anwendung kann mit dieser Plattform

realisiert werden. Der Aufwand ist in manchen Bereichen - z.B. Persistenz - aufgrund der unterschiedlichen Leistungsfähigkeit der Mobilgeräte umständlich. Ende letzten Jahres hat Sun bekannt gegeben, dass die Weiterentwicklung von J2ME eingestellt wird. Aufgrund der anhaltenden Leistungssteigerung der Geräte soll eine Umstellung auf J2SE stattfinden. Inwieweit die vorhandenen JSR s in J2SE integriert werden ist noch unklar. Die Migration wird noch einige Zeit beanspruchen. Für die Umsetzung macht es keinen Sinn auf "veraltete" Technologie - J2ME- zu setzen. Eine Entwicklung mit J2SE ist zum jetzigen Zeitpunkt wegen des nicht abgeschlossenen Migrationsprozesses ebenfalls keine Alternative.

Das .NET Framework von Microsoft ist ebenfalls am Markt etabliert. es stellt alle benötigten Komponenten für die Realisierung bereit. Ein großes Manko ist der Verbreitungsgrad. Dieser beschränkt sich auf Geräte mit Microsoft Betriebssystem, welches in aktuellen Smartphones selten zum Einsatz kommt. Im Bereich des PDA 's ist es etwas weiter verbreitet. Diese Bindung der Anwendung an wenige Geräte ist nicht wünschenswert, daher wird auf eine Umsetzung mit .NET verzichtet.

Bleibt als Alternative Android. Die von Google entwickelte Plattform befindet sich noch in den Kinderschuhen, hat dadurch aber den Vorteil keinen veralteten Code unterstützen zu müssen. Das System basiert auf einer speziellen Java Virtual Machine und ist nach dem MVC Prinzip realisiert. Einzelne Module ergänzen die Basis des Frameworks um weitere Funktionalitäten. Für die Entwicklung stellt Google bereits eine auf Eclipse basierende Entwicklungsumgebung bereit, ebenso wie eine Simulator zum Testen der Anwendungen.

4.3 Verwendbare Standards

Eine wesentliche Komponente stellt die Kommunikation zwischen den Geräten dar. Diese soll über eine definierte Schnittstelle stattfinden. Weiterhin soll die Darstellung auf möglichst vielen Geräten möglich sein. Um dies zu gewährleisten soll auf Standards zurückgegriffen werden. Zur Darstellung bietet sich HTML an, da die aktuellen Smartphones fast alle über einen Browser verfügen. Weiterhin besteht die Möglichkeit die Mozilla Engine Gecko in eigene Anwendungen zu integrieren. Ein weiterer Vorteil besteht in der Ausgereiftheit und der Anwendungsvielfalt im Bereich des HTML. So gibt es diverse Programme die sich auf eine templatebasierte Darstellung von Daten spezialisiert haben. Dadurch ist eine Personalisierung zu realisieren.

Für die Kommunikation eignet sich das URI Based Naming. Dies ist den Anwendern ebenfalls aus der WWW Welt bekannt und kann sowohl durch ihn als auch durch die Anwendung interpretiert werden. Auf dieser Basis lassen sich spezielle URL's als Sonderfunktionen deklarieren. Dazu zählt z.B. die Konfiguration der Geräte mittels der URL Device/Konfigure. An diese können URL Parameter angehängt werden, um Werte zu ändern. Die Anwendung hat dann die Aufgabe, diese Werte in für das betreffende Gerät verarbeitbare Befehle umzusetzen.

Weitere spezielle URL wären z.B. Trigger, die als virtueller Dienst in der Anwendung

hinterlegt werden können. Diese sorgen dafür, dass wenn ein bestimmtes Ereignis eintritt (Triggerevent) eine vom Benutzer definierte Aktion ausgelöst wird (Triggeraction). Mit Hilfe von Pseudogeräten lassen sich Aktionen gruppieren, sodass z.B. alle Heizungen beim Erreichen einer Zimmertemperatur von 24°C Grad heruntergefahren werden. Dadurch muss ein Triggerevent nur eine Aktion ausführen und nicht jede Heizung einzeln ansprechen. Das macht die Anwendung dann selbständig.

Für die Verwaltung der Berechtigungen wird ebenfalls eine Schnittstelle benötigt. Die Authorisierung soll in drei Stufen durchgeführt werden können. In der einfachsten Variante erlaubt ein Gerät generell den Zugriff oder verweigert diesen. Darüber liegt eine differenzierte Benutzerverwaltung, die Rechte pro Benutzer oder Gruppe ablegt. Diese Verwaltung kann auf dem Gerät selber sein oder delegiert werden. Die Delegation an andere Dienste ist somit die dritte Möglichkeit. Jedes Gerät muss mindestens den ersten Typ unterstützen, falls der zweite und dritte nicht möglich ist. Ein autorisierter Nutzer kann damit die Berechtigungen steuern. Dafür ist ebenfalls eine definierte URL einzusetzen, in etwa Device/Rights. . . Durch die Möglichkeit der Delegation von Rechteeanfragen ist eine beliebig komplexe Einsatzumgebung denkbar. Von einer Umsetzung der Dienstfindung soll vorerst Abstand genommen werden. Für die Umsetzung soll auf Android gebaut werden. Der junge Entwicklungsstand sollte dabei nicht hinderlich sein. Die Zukunft ist ebenso sichergestellt, da diverse Mobiltelefonhersteller dem Projekt beigetreten sind. Die Basis der vorhandenen Module nimmt viele Aspekte auf die aktuellen Softwareentwicklungsrichtlinien entsprechen. Da bereits das Telefonieren eine gleichberechtigte Anwendung auf dem Gerät darstellt, wird die Umgebung besser für Anwendungsentwicklungen geeignet sein, als ein proprietäres System, in das z.B. J2ME integriert wurde.

5 Risiken

In diesem Kapitel soll zur Vorbereitung der Masterarbeit ein Überblick über die Risiken gegeben werden.

Das größte Risiko bildet wie bei vielen Projekten die Zeit. Da die Masterarbeit auf sechs Monate begrenzt ist, soll ein Projektplan zu Beginn der Arbeiten erstellt werden. Dieser soll konkrete Projektphasen enthalten und festlegen, welche Komponenten in welcher Abfolge fertig gestellt werden sollen. Für eine grobe Vorgabe kann hierzu die Tabelle mit Anforderungen herangezogen werden. Anhand dieser sollen die Komponenten identifiziert und mit Zeitvorgaben versehen werden. Für die einzelnen Komponenten sollen parallel Test entwickelt werden, die zur Verifizierung der Funktionalität dienen.

Neben der eigentlichen Anwendung sollen noch Geräte, die Dienste anbieten, geschaffen werden. Diese dienen zur prototypischen Implementierung und Validierung der Anwendung. Sie können durchaus als nicht funktionale Dienste realisiert werden, müssen aber ein konkretes Verhalten simulieren können. Eine Schwierigkeit bei der Implementierung bildet die Kommunikationsschnittstelle. Da auf URL Based Naming zurückgegriffen wird, könnte ein Webserver mit entsprechender Logik zum Einsatz kommen. Eine vollständige Implementierung der Dienste ist dennoch wünschenswert.

Als weiteres Risiko ist die Android Plattform anzusehen. Durch den Alpha-Status der Entwicklung sind größere Änderungen an den Systemen nicht auszuschließen. Weiterhin kann es durch fehlerhafte Implementierungen der benötigten Androidkomponenten zu Verzögerungen kommen. Google schließt zudem Umstrukturierungen im Framework nicht aus.

Sollte die Entwicklung erfolgreich verlaufen, steht noch eine große Hürde am Ende. Da es noch keine Android-Hardware zu erwerben gibt, kann die Entwicklung lediglich am Simulator überprüft werden. Aus Erfahrungen im Bereich der J2ME Entwicklung hat sich ergeben, dass ein Gerät Code teilweise anders interpretiert oder durch die Ressourcenrestriktionen den Dienst komplett verweigert. Damit dies nicht zu einem zu hohen Risiko eskalieren kann, wird von einer Umsetzung auf konkrete Hardware erst einmal Abstand genommen. Dadurch rückt die Applikationsentwicklung mehr in den Fokus der Arbeit. Da Android Java basiert ist, entfällt ein Großteil des Einarbeitungsaufwandes. Vor Beginn der Programmierung soll das Konzept hinter Android verinnerlicht werden.

Soll die Anwendung später auf reale Hardware portiert werden, bleiben weitere Risiken bestehen. Ein entscheidender Punkt ist die Benutzerakzeptanz. Obwohl die Studie von Koskela und Väänänen-Vainio-Mattila (2004) gezeigt hat, dass das Smartphone ideal zur Steuerung im häuslichen Umfeld ist, muss die Anwendung intuitiv von den Benutzern zu bedienen sein. In dieser Phase sollen spätestens Usabilitytests durchgeführt werden. Um eine leichtere Migration zu anderen Bedienkonzepten zu ermöglichen, soll die Anwendung von vornherein darauf ausgelegt sein, neue Techniken leicht zu integrieren. Weiterhin soll an dem MVC Prinzip festgehalten werden, um die Logik sauber von der Darstellung zu trennen.

Eine weitere Frage stellt die Akzeptanz bei den Herstellern von Geräten dar. Es ist unklar,

ob sich diese an der Findung eines offenen Standards zur Steuerung ihrer Produkte beteiligen. Durch proprietäre Lösungen können sie Alleinstellungsmerkmale realisieren, die unter offenen Standards nicht möglich wären. Ebenso unklar sind die Kosten, die für die Erweiterung der Geräte, um die Schnittstellen zu integrieren, anfallen. Auf der anderen Seite wäre es vorstellbar, die Ansteuerung soweit zu abstrahieren, dass die Hersteller eigene Module in die Anwendung einbinden können, die der Steuerung ihrer Geräte dienen. Dies ist in etwa gleichzusetzen mit der Treiberentwicklung unter aktuellen Betriebssystemen.

6 Konklusion

Die Entwicklung einer Anwendung zur intuitiven personalisierten Steuerung der häuslichen Umgebung hat viel Potential. Die Konzentration der Entwicklung auf neue Plattformen bringt den Vorteil der besseren Integration und kann vom Aufschwung profitieren.

Aus technischer Sicht betrachtet, gibt es viele Forschungsgebiete in diesem Umfeld. So ist die automatische Dienstfindung und Konfiguration seit längerem in der Forschung. Die Kombination mit einem mobilen Gerät kann dabei nur positive Effekte haben, da der Nutzer dieses immer bei sich trägt und den Umgang gewohnt ist. Die Schwerpunkte der Personalisierung und Authorisierung sind Gebiete, die bisher nicht weit in die mobile Welt integriert sind. Hier besteht die Herausforderung darin, die Schnittstellen gut zu wählen. Die Authorisierung ist ein sensibles Thema, welches aufgrund der Vielfalt unterschiedlicher Geräte nicht trivial ist.

6.1 Kooperation

In der Entwicklungsphase der Anwendung soll eine enge Kooperation mit anderen Studenten stattfinden. Zu diesen gehören Jan-Peter Tutzschke, der ein Framework zur Kontextbehandlung mit Android realisieren will. Jan Schönherr plant sich der Peer to Peer Kommunikation von mobilen Geräten anzunehmen. Für die Dienstfindung gibt es Ansätze von Dennis Holatz und Sven Vollmer, die sicherlich als Vermittler zwischen dem mobilen Gerät und den Dienst Anbietern fungieren können.

Nach der Entwicklung der Anwendung gilt es Kooperationen mit der Wirtschaft einzugehen, um das Produkt an die dortigen Anforderungen anzupassen.

6.2 Einsatzmöglichkeiten

Die Steuerung bekannter Geräte in einer definierten Umgebung ist heutzutage bereits möglich. Dazu kommen oft Server- oder Heimautomatisierungslösungen zum Einsatz. Bringt man die Steuerung auf mobilen Geräten unter, ergeben sich viele neue Möglichkeiten. Verlässt man die definierte Umgebung und erweitert die Anwendung um eine dynamische Dienstsuche, können Dienste an jedem beliebigen Ort angeboten werden. Der Benutzer bekommt das Angebot anhand seiner Einstellungen identisch aufbereitet. Man stelle sich stationäre Systeme vor, die den Nutzer Tageszeitungen oder Musik verkaufen. Dies könnte im Vorbeigehen geschehen, ohne dass der Nutzer interagieren muss.

A Literaturverzeichnis

- [Cook und Das 2007] COOK, Diane J. ; DAS, Sajal K.: How smart are our environments? An updated look at the state of the art. In: *Pervasive Mob. Comput.* 3 (2007), Nr. 2, S. 53–73. – ISSN 1574-1192
- [Das und Cook 2006] DAS, Sajal K. ; COOK, Diane J.: Designing and Modeling Smart Environments (Invited Paper). In: *WOWMOM '06: Proceedings of the 2006 International Symposium on on World of Wireless, Mobile and Multimedia Networks*. Washington, DC, USA : IEEE Computer Society, 2006, S. 490–494. – ISBN 0-7695-2593-8
- [Dreyer 2008a] DREYER, Markus: *Pervasive Spine*. 2008. – Bericht INF-M3 PO
- [Dreyer 2008b] DREYER, Markus: *Your Home in Your Hand: Compare current work*. 2008. – Bericht INF-M3 AW2
- [Gamma u. a. 2005] GAMMA, E. ; HELM, R. ; JOHNSON, R. ; VLISSIDES, J.: *Design Patterns*. Addison-Wesley, 2005. – ISBN 0-201-63361-2
- [Johanson u. a. 2002] JOHANSON, Brad ; FOX, Armando ; WINOGRAD, Terry: The Interactive Workspaces project: experiences with ubiquitous computing rooms. In: *Pervasive Computing, IEEE Volume 1, Issue 2 (2002)*, S. 67–74. – URL <http://ieeexplore.ieee.org/Xplore/login.jsp?url=/iel5/7756/21806/01012339.pdf>
- [Johanson u. a. 2004] JOHANSON, Brad ; FOX, Armando ; WINOGRAD, Terry: *The Stanford Interactive Workspaces Project*. 2004. – URL <http://hci.stanford.edu/cstr/reports/2004-05.pdf>. – stand 2007-07-29
- [Koskela und Väänänen-Vainio-Mattila 2004] KOSKELA, Tiiu ; VÄÄNÄNEN-VAINIO-MATTILA, Kaisa: Evolution towards smart home environments: empirical evaluation of three user interfaces. In: *Personal Ubiquitous Comput.* 8 (2004), Nr. 3-4, S. 234–240. – ISSN 1617-4909
- [MIT] MIT: *Kerberos*. – URL web.mit.edu/Kerberos/. – stand 2008-01-18
- [Nakajima 11–14 July 2005] NAKAJIMA, T.: Personal coordination server: a system infrastructure for designing pleasurable experience. In: *Pervasive Services, 2005. ICPS '05. Proceedings. International Conference on (11-14 July 2005)*, S. 156–165
- [Nakajima und Satoh 2006] NAKAJIMA, Tatsuo ; SATOH, Ichiro: A software infrastructure for supporting spontaneous and personalized interaction in home computing environments. In: *Personal Ubiquitous Comput.* 10 (2006), Nr. 6, S. 379–391. – ISSN 1617-4909

[Park u. a. 2003] PARK, Sang H. ; WON, So H. ; LEE, Jong B. ; KIM, Sung W.: Smart home - digitally engineered domestic life. In: *Personal Ubiquitous Comput.* 7 (2003), Nr. 3-4, S. 189–196. – ISSN 1617-4909

[Tanenbau und van Steen 2007] TANENBAU, Andrew S. ; STEEN, Maarten van: *Distributed Systems - Principles and Paradigms*. Pearson, 2007. – ISBN 0-13-239227-5

[Waseda-University] WASEDA-UNIVERSITY: *Distributed and Ubiquitous Computing*. – URL www.dcl.info.waseda.ac.jp. – stand 2008-01-18

B Glossar

Android Auf Linux basierende Java Plattform für Mobiltelefone.

Augenblickliche Interaktion Aktionen, die in einer sofortigen Reaktion eines Gerätes münden (Instant Action).

Eclipse Open Source Anwendungsentwicklungsumgebung.

Gecko Layout Engine des Mozilla Projectes.

J2ME Sun Java Micro Edition.

J2SE Sun Java Standard Edition.

JSR Java Specification Request, sind Beschreibungen für die endgültigen Java Spezifikationen.

Mustergesteuert Aktionen, die automatisch bei bestimmten, vorher definierten Ereignissen ausgeführt werden (Pattern Control).

MVC Model View Controller, Architektur zur Softwareentwicklung. Dabei werden das Modell, die Präsentation und die Steuerung voneinander getrennt.

PDA Personal Digital Assistant: Ein kleiner portabler Computer, der den Benutzer bei alltäglichen Aufgaben unterstützt.

Webservice Standardisiertes Verfahren zum Zugriff auf Ressourcen im Netzwerk.

X10 Ein Standard zur Kommunikation von Geräten zur Heimautomatisierung.