

LDAP Contact Sync: Kontaktsynchronisation mit LDAP Kontaktverzeichnissen für Android

Android Praktikum
Sommersemester 2011

Projektteam

Till Lorentzen	Daniel Fricke	Christopher Gerloff
Tel: 0176 70186920	Tel: 0160 96455733	Tel: 0174 1822440
t.lorentzen@tu-bs.de	daniel.fricke@gmx.de	c.gerloff@tu-bs.de
Teamleitung	Backend	GUI

Motivation und Ausgangssituation

Kontakte sind einer der zentralen und essentiellen Datenbestände auf einem Mobiltelefon. Daher ist eine Synchronisation mit bereits vorhandenen Kontaktdatenbanken sehr wichtig für den schnellen Abgleich mit anderen etablierten Desktopprogrammen. Da LDAP Datenbanken sehr weit verbreitet sind, aber Android von Haus aus keine Möglichkeit bietet auf diese zuzugreifen, wollen wir diesen Mangel beheben. Wir wollen Contentprovider für eine LDAP Adressdatenbank implementieren und die Synchronisation zwischen Android Kontakten mit der LDAP Datenbank ermöglichen.

Braunschweig, 08.04.2011

Till Lorentzen _____

Daniel Fricke _____

Christopher Gerloff _____



 aggeber:

<Name> _____

eher "einen Contentprovider" ?

Schreibt meinen Namen oder "Android
Lab"

Tech Details: Wahrscheinlich reicht ein Hintergrundprozess für beides? Und man sollte das Auto-Sync vom User an und abschalten lassen können, bzw Zeitintervall festlegen in dem geprüft wird. (Background Prozes kostet ja auch Energie und vielleicht will das nicht jeder User)

Ziel

Eine Integration von Kontakten aus einer LDAP Datenbank in das Kontaktbuch und deren Synchronisation.

Anforderungen

Unser Programm soll folgende Funktionalitäten haben:

- Verbindung zu OpenLDAP Server herstellen
- Kontakte des verbundenen OpenLDAP Servers durchsuchen
- Kontakte einzeln oder mittels Filter in das Kontaktbuch des Mobiltelefons importieren
- Importierte Kontakte entweder hinzufügen, oder mit vorhandenen Kontakten abgleichen und zusammenfügen
- Kontakte aus dem Mobiltelefon in die LDAP Datenbank exportieren
- Bei Änderung eines Kontakts auf dem LDAP Server oder dem Telefon, diese Änderung auf dem entsprechend anderen Datenbestand aktualisieren.
- Sollten nur Leserechte auf der Datenbank bestehen, ist nur die eine Richtung der Synchronisation möglich

Lösungsweg

Zuerst muss eine Verbindung zu einem, oder mehreren LDAP Server erstellt werden. Dann sollten die Kontaktdaten auf dem LDAP Server mittels Contentprovider allen Anwendungen auf dem Telefon verfügbar gemacht werden. Auf der anderen Seite sollen Activities erstellt werden, welche sich in das Adressbuch integrieren, um einen Kontakt zu importieren, sowie zu exportieren.



Ein Hintergrundprozess soll versuchen, bei Änderungen an einem Kontakt, der seinen Ursprung im LDAP hatte, oder mit einem aus dem LDAP Verzeichnis zusammengeführten, diesen Kontakt zu synchronisieren. Bei Konflikten soll der Nutzer später entscheiden können, welche Konfliktlösungsstrategie angewandt werden soll. Ein weiterer Hintergrundprozess sollte Änderungen auf dem LDAP Server beobachten und bei serverseitiger Aktualisierung eines importierten Kontakts diese Änderungen auch auf das Mobiltelefon übertragen.

Projektplan

Androidprojekt anlegen und Bibliotheken Recherche	04.04.11	18.04.11
Test der JLDAP Bibliothek	04.04.11	15.04.11
OpenLDAP Server Installation	04.04.11	15.04.11
Junit Tests für eigene Klassen schreiben	04.04.11	28.06.11
Erstellung 3-SP	05.04.11	13.04.11
GUI Prototypentwurf Graphisch	13.04.11	22.04.11
GUI Prototypentwurf Programmatisch	22.04.11	02.05.11
Erste Tests von Import und Export von Kontakten	18.04.11	15.05.11
Review 1	16.05.11	17.05.11
Merge Prozesse implementieren	17.05.11	30.05.11
Synchronisationsprozess implementieren	17.05.11	11.06.11
Backgroundservice implementieren	17.05.11	11.06.11
App, Funktionen & GUI verfeinern	17.05.11	18.06.11
Testenszenarios entwerfen und auswerten	11.06.11	26.06.11
Review 2	27.06.11	28.06.11
Abschlusspräsentation erstellen	29.06.11	11.07.11
Projektabschluss		13.07.11

Potentielle Hindernisse und Maßnahmen

Integrationsprobleme in die Standard Kontakte des Android Betriebssystems kann entstehen, falls die Views innerhalb des Kontaktbuches nicht geändert oder erweitert werden können. Dann müssen diese Views in die extra App integriert werden.



Hintergrundprozesse könnten viel Strom verbrauchen, falls zu oft synchronisiert wird. Das könnte mit manuellem Synchronisationsprozess behoben werden. Der Nutzer sollte zwischen automatischem und manuellem Prozess wählen dürfen.

Änderungen in der LDAP Datenbank können evtl. nicht erkannt werden, wenn der LDAP Server über Änderungen nicht informieren kann. Evtl. kann dieses Problem mit einem Push Server gelöst werden, wobei dieser nicht essentieller Teil des Projektes sein wird, da er nicht auf dem Android OS laufen kann.

OpenLDAP Bibliothek für JAVA könnte nicht kompatibel zu Android sein. Kann gelöst werden, indem andere Bibliotheken benutzt werden könnten, oder auch C++ Bibliotheken evtl. als letzte Möglichkeit überprüft werden müssten.

Ah hier steht es. Würde es eher hier raus nehmen und oben mit reinbauen. Denn eigentlich weiß man ja jetzt schon dass es eine gute Idee ist das konfigurierbar zu machen