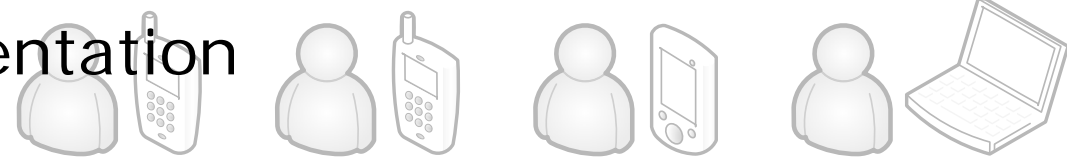


Softwareprojekt Mobilkommunikation

Abschlusspräsentation



- Introduction / Background (by L. AiQuan)
 - Mobile Phones, Android, Use Cases, ...
- Architecture (by Janos K.)
 - Components, Interfaces, ...
- Applications (by Carlos D.)
 - Required Components, Setup, ...
 - GUI Elements, Interaction, ...
- Conclusion (by L. AiQuan)
 - State of the Project
 - Future Work

- Mobilkommunikation

Def: bezeichnet Sprach- oder Datenkommunikation mittels mobiler, drahtloser Endgeräte

Möglichkeiten:

- Netzwerk mit fester Infrastruktur
- Ad hoc Netzwerk: ein Netz ohne Infrastruktur

- Android als Endgeräte bei dem Projekt

Neu auf dem Markt aber schnell verbreitet

- Bis Weihnachten 13 Android Phones angekündigt
- Tablet PCs
- HP testet Android für PCS
- Asus plant Android EEEPcs
- Archos einen Android Media Player

- Mobilkommunikation Projekt SS09

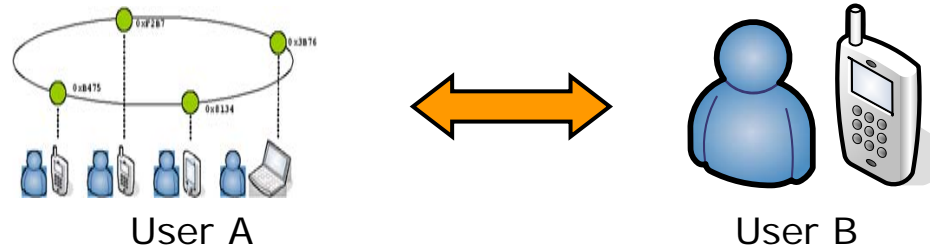
Ziel: Für die beide Mobilkommunikationsmöglichkeiten jeweils eine Lösung/ einen Einsatz entwickeln und das Prinzip von Mobilkommunikation verstehen

- Client/Server

Use Case: Adresse Buch

- Konkrete Use Case Beschreibung sehen die nächste Folien aus der ersten Präsentation als das Projekt begann.

Use Case - (Address Book)



Setup:

- User A gives ID# of his user profile (stored on PDA) to user B
- User B adds user A's profile to address book of his mobile phone by entering ID#
- User A authorizes user B's access to his profile
- User A's profile is now part of address book on user B's mobile phone

Benefits:

- If user A changes his contact data in his profile, these changes propagate automatically to user B's address book.
- In case user B's mobile phone is switched off or disconnected, the data is synchronized at a later point.

- Peer to Peer Netzwerk

Def:

In einem reinen Peer-to-Peer-Netz sind alle Knoten gleichberechtigt und können sowohl Dienste in Anspruch nehmen als auch Dienste zur Verfügung stellen. Kernkomponente in einer Peer-to-Peer-Architektur ist das Overlay-Netzwerk, welches den Peers im Netzwerk die Funktionen Lookup und Suche zur Verfügung stellt.

Funktionen:

- Lookup: Mit der Lookup-Operation können Peers im Netzwerk diejenigen Peers identifizieren, die für eine bestimmte Objektkennung (Object-ID) zuständig sind
- Suche: Mittels der Such-Operation können die Peers nach Objekten im Netzwerk suchen,

Die im Projekt entwickelten Applicationen

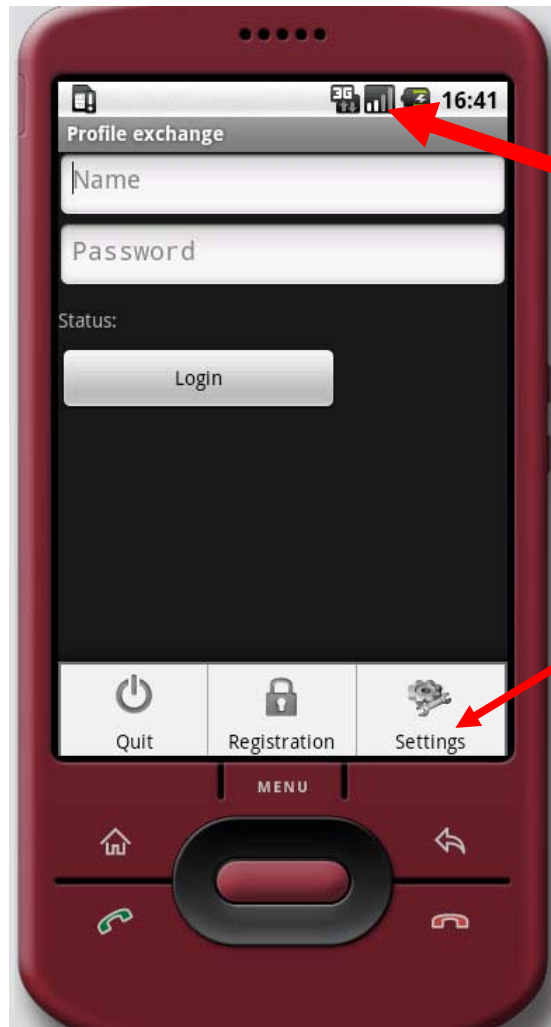
- Profilverwaltung (Address Book)
Auf Android Plattform, Ad hoc Netz

- Karten und Positionierung
Karte anzeigen lassen
Position einer Person ermitteln und auf der Karte zeigen lassen

Architecture

[TODO]

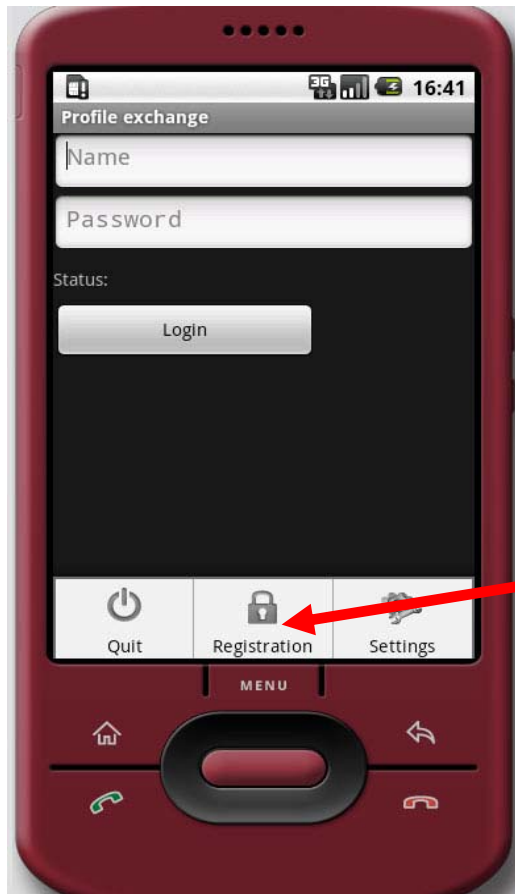
Applications



- The application needs to have network access, through the emulator or a direct connection.
- To connect to the network the user has to know a server that provides the service.
- IP or hostname and ports are required

Registration and login process

The first screen shown manages both processes with the same elements:



- Integrated registration process
- Identification through an unique username
- Log-in process will fail for unregistered users
- **Can not** interact with the main application functionalities without being logged

Switching between register and login using MENU button.



Android MENU functionality:

- Basic "right click" options
- Usually hiding secondary options
- Remember it's there !



- Tab oriented, with one tab per task, very user-friendly: no mixed functionalities.
- Non-aggressive notifications or uses:
 - No workflow interruption.
 - No automatic actions taken.
- Application can run in the background.
- Secondary functions hidden under the MENU button.

Profile tab

First screen presented to the user, manages own profile information, locally stored:

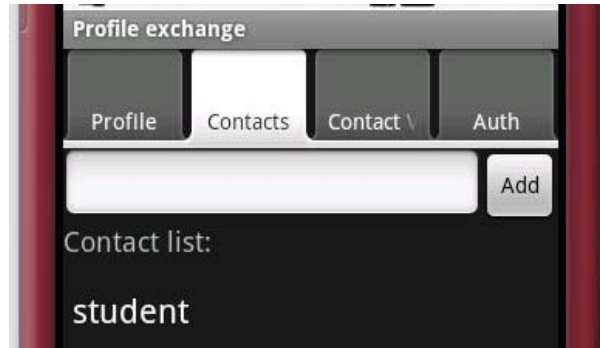


No need to be always synchronized with remote server. Editing can be done at any time.

Information is also stored server side but only on user-request by click on the "Save" button.

Two tabs involved:

- Plain contact list (by name)
- Contact view: detailed view

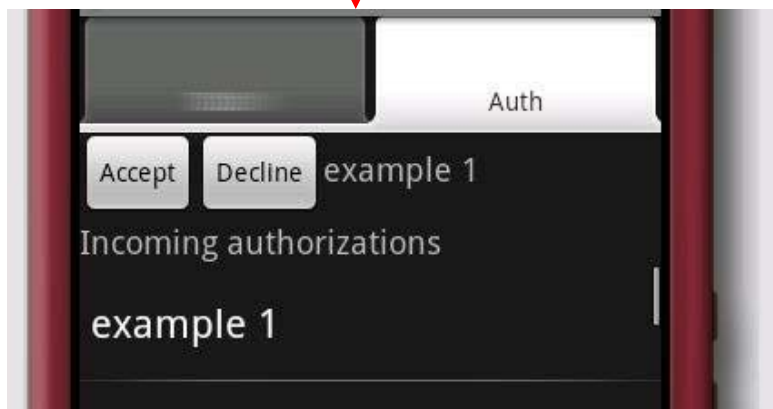
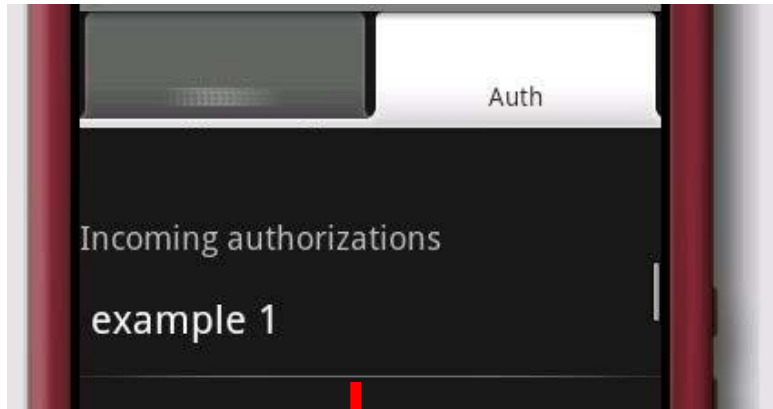


Adding contacts needs no extra interface, the contact's unique username is provided and the request is sent to the server.

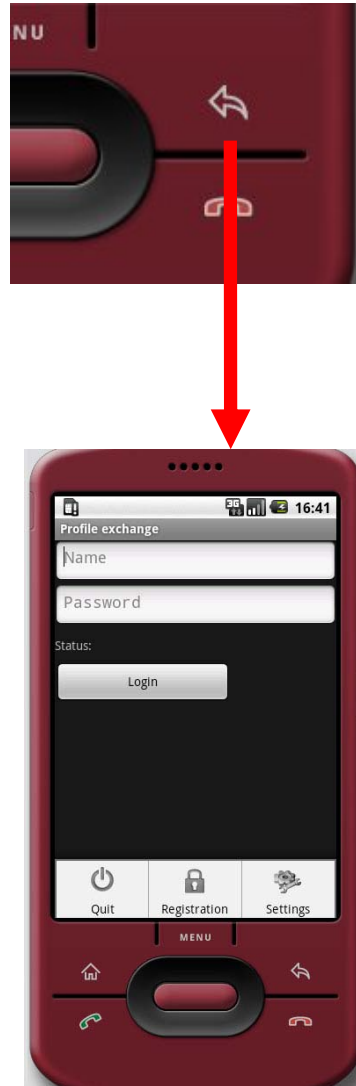
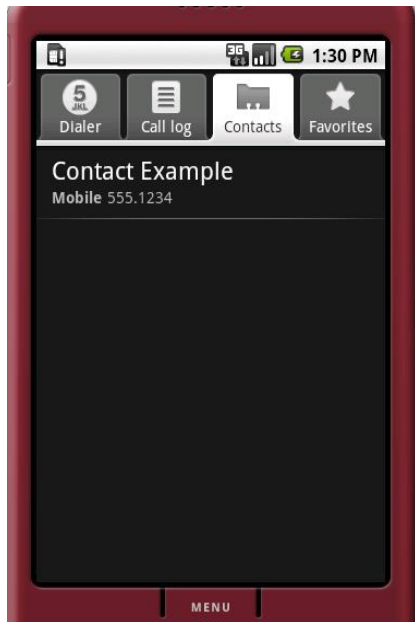
Deleting a contact is part of the user options. A complete clear can be also done using the MENU button menu.

Once a contact is clicked the view is switched to the detailed view tab which holds **user state** and **last update date**.

Contacts are stored in the OS contact list once they are confirmed by the remote user, name and phone number are stored to be able to call them as regular phone contacts.



- To be non-aggressive towards the user, when an incoming request is registered it is depicted in a pending requests list.
- The user can authorize or decline a remote request at anytime in a 2 clicks process.
- Once a contact has been **accepted** it is added to the local and OS contact list.
- If the user **declines** the request, the server is notified and no further actions are taken.



- Local OS storage of contacts (see before).
- MENU button context menu holds different options.
- Application can run in the background using Android's "back" button, but user needs to log-in again once restored.

Conclusion

- Zustand von dem Projekt
 - Client/Server
 - Alles funktioniert wie geplant
 - Verbessern: ein paar fehlerbehandelnde Nachrichten implementieren um das Prototype zu verfeinern.
 - Peer to Peer
 - Profilverwaltung
 - Was funktioniert was nicht, was noch fehlt
 - Karten und Positionierung
 - Das Rendering der Kartendaten funktioniert schon gut
 - Die Darstellung der Karte geschieht in Abhängigkeit vom Standort
 - Der Standort einer Person kann über den Server/die DHT ermittelt werden
 - Der Aufruf dieser Aktivität geschieht noch nicht durch das Hauptprogramm

- Wie wir mit dem Projekt weiter machen

Freie Diskussion

Danke!