Requirements for Tourist Application

The aim of the Mobilis tourist application is to enable tourists to travel in group and share experiences with group members. It aims also to guaranty each group member a high degree of flexibility and freedom. Towards this end, the application should support group-awareness, facilitate the seamless exchange (sharing) of context and experience, and the unobtrusive delivery of messages. This document identifies core services which are required to fulfill the aims of the application. The basic requirements of each service are also listed.

Contents

1. Application Overview
2. Architecture
3. Functional Requirements for the Proposed Application
   1. Group Management Service (GMS)
   2. User Communication Service
   3. Group Location Awareness Service
   4. Map Service (MP)
   5. Fun Flag Service (FFS)
   6. General Purpose Notification Service
   7. Application Main Adaptations
   8. Context Service (CS)
   9. Context Providers (CP)
4. Requirements for the first prototype

Application Overview

On each Mobilis platform runs a Group Management Service (GMS) so that individual members can define the members of their group and specify their relationship with each one of them. In case of the absence of explicit specification, default relationship will be established. The default relationship will look like the relationship of commonly traveling tourists in group. The Figure below shows how an application snapshot could look like.
Once a relationship is defined, the user has the right to modify a relationship at any time. He has also the right to define the type of content and context information individual members should access or send. Moreover, at any given time the GMS is reconfigurable. Besides group management, the GMS also access additional services, such as the Map Service (MS), Context Service (CS) and the Fun-Flag Service (FFS), to enrich the user with context and multimedia information. For example, the GMS accesses the CS and the MS in order to display at various granularities the whereabouts of group members.

**Architecture**

The figure below displays a conceptual architecture for the Mobilis tourist application. The application is a composition of several underlying services each of which provides the application with certain functionalities. The application also implements application specific functionalities which determines

1. The user interface;
2. The way services interact with the application and with each other; and
3. Additional functionalities that cannot be provided with existing services.

The architecture consists of four basic service layers: the higher-level service layer, the lower-level service layer, the operating systems (and VM) layer and the physical (hardware) layer.
Functional Requirements for the Proposed Application

**Group Management Service (GMS)**

1. Adding and removing members as well as modifying the relationship type a user has with other members
2. Manage users (add/update/delete)
3. Manage groups (add/update/delete groups, add users to groups)
4. Defining relationship to other group members (maybe role-based)
5. Authentication of users
6. Authorization (managing policies for the access to content and context information)
7. Buddy activity/status awareness:
   - A user must be notified about changes in buddy status, in terms of activity, presence, communication interest, etc.
   - Common tourism interest (to gather together and, eventually, form a small group to visit places, etc)
   - Sharing of tourism interest with others who haven’t seen a given place
   - Common goal other than visiting places
   - Sharing costs during a tourism
   - Relaying messages in case of a MANET
   - The specific case of buddy location awareness is described in the requirement Group Location Awareness Service.
8. Define/Consult group schedules (to specify better)
9. Define/Consult/Accommodate group tasks

**User Communication Service**

Users must be able to communicate through:

- synchronous (instant) messages (text, special-defined messages such as Hold On...)
- asynchronous messages (text and e-mail)
- media sharing
- VoIP
- SMS
- Built-in camera
- Group communication (e.g., for sharing experiences and other information with related people).

**Group Location Awareness Service**

1. Display of the position of the other group members, with features:
   1. user’s position updates should be automatic (i.e. without input from the user)
   2. position updates should be displayed at other group members’ devices with minimal delay (within 30-60 seconds)
   3. when a device becomes disconnected (or unavailable, e.g. insufficient energy) the last seen position displayed at the other’s devices should be shown with a characteristic color indicating it staleness

   **Rationale:** users want reliable and almost real-time information of the other’s position

2. Seamless integration with other communication services, with following features:
   1. by selecting a given user icon on the map, his/her contact (e.g. phone number or Email address) should appear on the screen, allowing a one-click-only call/connection with the selected user

   **Rationale:** at some occasions, users want to communicate directly with other users, and should not need to switch the applications

3. Explicit specification of the access right of individual members to shared information
4. Defining the types of content a user would like to receive from member groups

5. Reconfigurable

**Map Service (MP)**

**Goal:** The MS is an adaptive map service that enables the GMS to trace the group members when they are indoors, outdoors or distributed in indoors and outdoors.

1. Should work in and outdoors, and the switching should be automatic and seamless.

   **Rationale:** *The user really doesn't care which positioning technology is being used*

2. Display of Maps with following features:

   1. Zoom in/out
   2. Automatic crop of the map showing the surroundings of the user (as he moves)
   3. Highlight of the buildings/monuments/landmarks of interest
   4. Automatic loading of contextualized maps

   **Rationale:** *Maps are the most important tools for tourists in an unknown site/city*

3. Display of the position of the other group members, with features:

   1. User's position updates should be automatic (i.e. without input from the user)
   2. Position updates should be displayed at other group members' devices with minimal delay (within 30-60 seconds)
   3. When a device becomes disconnected (or unavailable, e.g. insufficient energy) the last seen position displayed at the other's devices should be shown with a characteristic color indicating it staleness
   4. It should enable the user to know at various granularities the whereabouts of the group members;

   **Rationale:** *Users want reliable and almost real-time information of the other's position*

4. The map service interacts with the GMS so that the GMS can make the whereabouts of members known to the user

5. Presentation of fun flags (see requirement *Fun Flag Service*)

6. Calculate routes and distances between two points

7. Providing information of places (e.g. museums, touristic places)

8. Privacy-related requirements

   1. Location tracking must be constrained by the member's access policy. For example, a member may not mind to share with the other members the streets address where he
is now without actually letting then know the exact location of the place where he is. This applies to indoors as well.

2. An **invisible** function, with following features:
   1. by means of an easy-to-operate button or menu item
   2. display of the visible/invisible status on own device
   3. display of the visible/invisible status to other users (e.g. a special color of the last seen position)

   **Rationale:** *the user may not wish to share his/her location with the group all the time*

**Fun Flag Service (FFS)**

The fun flag is useful to share rich multimedia information with group members.

1. Ability to create a new fun-flag and place it on any position on the map (even on a different than the current position), with following features:
   1. Any user may add one or more contributions to a fun flag, as well as a fun score, which is used to calculate a mean fun score. A contribution can be a photo, a video, a text or a voice message, and has a timestamp and the ID of its author.
   2. Whenever a user clicks on a flag, he/she is able to view/hear/read all the contributions associated to the flag, and may directly enter his own contribution.
   3. A user must not edit or remove other’s contributions or flags. They are persisted, and only the administrator can remove/edit contributions and flags.
   4. Any new flag or contribution should be visible/available for access immediately to all users (latency of less than 2 minutes)
   5. After a period of disconnection or non-availability (either of the author or a viewer of contribution/flag), all local and remote updates of contributions and flags must be delivered to all the users.

   **Rationale:** *in spite of intermittent connectivity, flags and contributions should be available to all the interested user. However, a reputation mechanism will not be supported.*

2. The SS makes a fun-flag folder available to the user to save sight related multimedia information.

3. Each time the user encounter interesting information, he haves it in the folder together with time and location information

4. Either fun-flags are pushed to the members or members can set a query request and collect information

5. The FFS uses the GMS and the MS to make multimedia data available to the group members

**General Purpose Notification Service**

1. Register & remove general purpose interest expressions, with following features:
   1. Ability to easily register his/her interest in different sorts of events, as well as the desired notification mechanism (ring-mode, vibrator-mode, display flashing, etc.)
   2. Interest registrations should be available for many different types of events of any service, e.g. "any new contribution by userX", "any new fun flag in regionY", "any movement of {userW,userZ}", "user became disconnected", "userX got more than 2 km far from userZ", "event X has happened", "place X is near".
3. Notifications should be reliably delivered and in causal order.
4. After a period of disconnection or non-availability, non-delivered notifications should be delivered immediately.

**Rationale:** The tourist has many interesting things to see, therefore he/she does not want to always keep an eye on his/her device. Therefore, an asynchronous notification service is useful.

**Application Main Adaptations**

Application must transmit messages according to different variables (network connectivity, amount of energy available at the user's site, activity the user is performing at that moment, cost, security, etc)

**Context Service (CS)**

The CS makes all interesting context regarding the group available to a user or a service. To this end, it should enable the user

1. To declare the types of context the user or a service is interested to
2. Model a user's relationship with other groups
3. Subscribe to several distributed context providers
4. The CS interacts with the GMS to display context information to the user

2. It enables the GMS to trace the group members when they are indoors, outdoors or distributed in indoors and outdoors.
3. It should enable the user to know at various granularities the whereabouts of the group members; this granularity is, however, constrained by the member's access policy, for example, a member may not mind to share with the other members the streets address where he is now without actually letting then know the exact location of the place where he is. This applies to indoors as well.
4. The map service interacts with the GMS so that the GMS can make the whereabouts of members known to the user

**Context Providers (CP)**

1. A CP abstracts a context source, which can be a GPS or a calendar.
2. It implements a query and subscription request processors to make timely context available to the CS.

**Requirements for the first prototype**

1. Technologies to be considered
   1. PDA-like devices
   2. WLAN interfaces
   3. Bluetooth interfaces (tentative)
   4. GPS or another location-related solution
   5. Java and J2ME
   6. Jxta and XMPP
   7. Windows Mobile + Windows XP
   8. JNI + Socket communication between native code and Java
   9. OSGi
2. Main concepts
   1. Group vs Buddy list
2. Role of each person
3. Spatial (geo-location) and temporal info
4. Activity and goal
5. Type of data to be shared
6. Spatial

3. Main Features

1. Managing a closed group of travelers (TUD)
   1. create group members
   2. create a group
2. Awareness in the group (TUD)
   1. sharing information about group members with all other members
      1. position, activity, connectivity status, willingness of communication
      2. automatically and constantly available after login
      3. each awareness information is time-stamped
3. Positioning of devices (PUC)
   1. GPS/WLAN based positioning
      1. geo-coordinates, semantic locations, mapping between both
4. Activity and Willingness to communicate (UFMG)
   1. user selects on activity out of a list with predefined activities
5. Connectivity (UFMG)
   1. captured from the network card (Wifi)
      1. bandwidth, delay, jitter, ...
6. Devices/Group members (UFMG)
   1. at least three PDAs/users
   2. at least two access points
7. Architecture (PUC, UFMG, TUD)
   1. client/server
   2. 3 clients and 1 server
8. Collaboration (TUD)
   1. text-based messages
   2. sharing of images (files) simple upload and download from server
9. Supported User mobility
   1. walking speed
   2. movements limited to a defined area (defined by map)
10. Adaptation (PUC)
    1. group member moves from indoor to outdoor and back (change of positioning system)
    2. Content adaptation at client and server side (different variants)
    3. dynamic service installation (image adaptation)

4. User Interface (UFMG, PUC)

1. User interface for the Prototype
   1. A map for visualizing locations of all group members
   2. based on google maps
   3. users with unknown locations are marked with red markers at the last known (initial) position
   4. users with known position are shown with green markers
   5. users which are indoor as shown as markers in the map (more precise location information based on awareness info.)
   6. awareness information is visualized as extended marker info.
2. Configuration Tab
   1. Awareness on/off
   2. controls for activity and willingness to communicate
3. List of group members tab (alternative view of group)
   1. all group members in a list
   2. awareness information with items and text
4. Tab for Chatting
   1. selection of communication partner?
   2. 2 text fields
5. Shared Files Tab
6. list of files
   1. add files to share