

Mobile Services Exam Projects



F. Ricci
2010/2011

Application Domains

- ❑ **For sale-based organizations:** provide information to their salespeople in the field, allowing them to bring up live pricing and estimates
- ❑ **For health-care orgs:** provide access to, formularies, policies, procedures, patient charts at the patient's bedside
- ❑ **For real estate:** provide potential buyers with live listings of similar properties nearby, information about the house, school information, average home values nearby, while touring a property
- ❑ **For local government:** increase awareness and community participation, to elicit feedback on public projects or weight important issues facing the community
- ❑ **For retailers:** provide instant points of purchase, provide ongoing customer support, link to online reviews, while the customer is holding a product in her hand.

Decide Together a TV Program

□ **Functions**

- Two or more people (e.g., a family) should be able to search in a TV program list
- Search program by keywords or metadata (genre)
- Browse detailed information on the program (plot, pictures, etc.)
- Select one program and make a proposal to your peers
- They can reply with 'yes' or make a counter offer
- Design a process that will "converge", i.e., they reach a common choice
- Store the selected program in a local database (all the peers)

□ **Implementation details**

- Find a list of TV programs from the web or build your own sample db
- Communicate with your peers with SMS (or sockets, if you have wifi).

Rooms Management

□ **Functions**

- There are some rooms – with various features - that are available on time slots
- Users can ask the system to reserve a room for them with certain particular features (capacity, projector, blackboard, etc.)
- Search for available rooms
- The system offers the rooms better suited for the request
- The user can accept or reject the system offer

□ **Implementation details**

- Implement everything on mobile devices – also the server with all the rooms data
- Store the rooms schedule in a record store
- Interacts with clients with either via socket, or http, or SMS.

News Recommendations

□ Functions

- Download a list of News from a RSS feed (e.g. BBC http://newsrss.bbc.co.uk/rss/newsonline_uk_edition/technology/rss.xml (better: a source of local news for Bolzano))
- Browse the result list
- Browse detailed information about the news
- Select a news and store it in a local store (organize the news in categories)
- Implement a function that **suggests** some items based on the similarity to some of the news in your local store

□ Implementation details

- Use an existing RSS feed
- Parse the XML using an existing library like kXML
- Exploit the RecordStore library

Task List Alert

□ **Functions**

- Imagine a **particular** Field Force Automation situation and design the application for that scenario: e.g., fuel distribution, parcel delivery, care giver ...
- Add, delete, modify a task (and its features)
- Add, delete, modify a task category
- View tasks filtering by date, category, keyword, location (according to the application)
- Build an alert function that alert you
 1. when a task is due
 2. you are in a good situation to perform the task (e.g., approaching the right location).

□ **Implementation**

- Use the MIDP record Store, and the location API

Care Giver

□ **Functions**

- Service Request (an ill person in need of assistance)
 - User sends help request by pressing a key
 - She gets a response from the care giver (read and confirm receipt)
- Receive directions from care giver
 - User receives an alert message (e.g. "take the pills")
 - The alert is repeated if no confirmation is given
 - User acknowledges the receipt and the acceptance to the caller
 - After a timeout the caller is warned that the receiver has not acknowledge the request

□ **Implementation**

- Use SMS for exchanging information between the care giver and the user
- The MIDlet must listening for incoming SMS and make the interaction simple and easy to manage.

Route Geo Tagging

□ **Functions**

- Record a sequence of positions and related notes (picture, comment, features) during a route (RouteStage) – imagine a specific application scenario (e.g., annotation of a bike route or mountain path)
- Upload the RouteStages to an [Exist](#) DB and keep them organized in an Itinerary
- Retrieve the itineraries that are closer to you and with features that you are interested in (length, slope, terrain..)

□ **Implementation**

- Position can be obtained using the location API
- Use Exist server APIs for uploading and downloading RouteStages
- The RouteStages are organized in annotated collections

Around me

□ **Functions**

- Download a list of geotagged Hotels, Restaurants, Attractions from an XML Feed that will be provided
- Retrieve the elements that are closer to the current location of the user or to a position specified manually
- Select elements and store them in a local store - organize the selected elements in the calendar
- Implement a function that **suggests** additional attractions and restaurants according to proximity and free slots in the calendar (taking into account opening hours)

□ **Implementation details**

- Parse the XML using an existing library like kXML
- Use Exist server APIs for uploading and downloading XML data

Orienteering

□ **Functions**

- Implement a game where a user get a mission on the mobile phone
- The mission consists in reaching a certain number of targets
- When you get a target you get the description of the next target (e.g., find a location where there is x)
- The mobile must log your actions
- At the end of the game the mobile debriefs the game
- Send the mission execution log to another user

□ **Implementation**

- Position can be obtained using the location API
- The missions are stored locally by the application and randomly assigned to the user.

Domotics (Home Automation)

□ **Functions**

- Control the state of your heating system (raise the temperature in one room, cover/uncover the solar panels, start-up/shut-down the furnace, ...)
- Receive messages from your heating telling you how much you are spending
- Define rules that switch-on/off based on your location
<http://www.technologyreview.com/blog/mimssbits/25752/?nlid=3507>

□ **Implementation**

- Build a nice GUI for this application (simple!)
- Use SMS or sockets connections with push
- Simulates the events in the furnace and the actuation of commands
- Imagine that in the house there is another "phone" that communicate with your mobile.

Generalities

- ❑ The project should require more or less 40 hours
- ❑ The project “deliverables” includes:
 - The code (one zip file): must be deployable in the Java ME SDK
 - If you need a server component – implement and upload it as you like, **but should be accessible from FUB**
 - One text file (readme.txt) describing all the required steps to install and run the program on the SDK
 - One document (pdf file, **less than 3000 words**) describing:
 - ❑ The system functions and the human computer interaction: what is doing and how a typical interaction proceeds (use snapshots from the GUI)
 - ❑ The structure of the code (classes and files): describe the role of each class and the main methods
 - ❑ The major technical problems that have been tackled and how these have been solved.

About the grade

- ❑ I will evaluate the following elements:
 - **Usability of the system** (no errors, no strange commands, easy to learn, enjoy using the system)
 - System functions coverage – with respect to the project description
 - Quality of the presentation
 - Quality and completeness of the report
 - Deployment and running - how fast I will deploy and run the system on my PC without errors
- ❑ Present your project at the last class if you want to pass at the winter session
- ❑ The presentation must be contained in 15 mins and must illustrate the same points described in the report
- ❑ **If you reuse existing code you must clearly write what code you used and what have you done on top of that!**

Next steps

- Build a group of 2 people or work alone
- Select a project among those described here
 - Or propose some changes to one of those
 - or write down a project description in a similar way
- Send an email to me (as soon as possible) with the following data
 - Group members: names and id numbers
 - Selected project: if it is not one of those suggested include the project description, or if you propose some changes (new functions) please describe them in the email.