

AAL-2011-4-078



## Deliverable D-2.2

# MOBILE.OLD Specifications and Validation Plan

<b>Deliverable Type:</b> PU*
<b>Nature of the Deliverable:</b> R**
<b>Date:</b> 30/01/13
<b>Distribution:</b> WP2
<b>Code:</b> MdO_T2.3_T2.4_SEPRO_D-2.2_300113
<b>Editor:</b> SEPRO
<b>Contributors:</b> ALL

\* **Deliverable Type:** PU= Public, RE= Restricted to a group specified by the Consortium, PP= Restricted to other program participants (including the Commission services), CO= Confidential, only for members of the Consortium (including the Commission services)

\*\* **Nature of the Deliverable:** P= Prototype, R= Report, S= Specification, T= Tool, O= Other

**Abstract:** This document represents the deliverable D2.2 – MOBILE.OLD Specifications and Validation Plan. The aim of this deliverable is to select and specify the services to be implemented, to select the equipment and the SW technologies required for the implementation and deployment of the services, to specify the MOBILE.OLD network, Application Server and Smart Phone SW architecture and to define the test and validation plan.

© Copyright by the MOBILE.OLD Consortium.  
The MOBILE.OLD Consortium consists of:

Participant no.	Participant organisation name	Country
1 (Coordinator)	LIFEtool gemeinnuetzige GmbH	Austria
2	National Foundation for the Elderly	The Netherlands
3	Madrid Health Cluster	Spain
4	Ana Aslan International Foundation	Romania
5	SIEMENS PROGRAM AND SYSTEM ENGINEERING SRL	Romania
6	SEPROTRONIC GmbH	Germany
7	SAFEVIEW	Spain
8	BLUE POINT IT SOLUTIONS	Romania
9	AdvTEC Ltd.	United Kingdom
10	Upper Austria University of Applied Sciences	Austria

This page has been intentionally left blank.

# Table of Contents

- 1. Introduction..... 10
  - 1.1 Scope ..... 11
  - 1.2 Deliverable Structure ..... 11
  - 1.3 Methodology ..... 12
- 2. MOBILE.OLD Network Architecture ..... 13
- 3. Selection of the services to be implemented..... 15
- 4. MOBILE.OLD Services Specification..... 17
  - 4.1 Personalized Transportation News Headlines Service ..... 17
    - 4.1.1 Service Overview ..... 17
    - 4.1.2 Service Configuration ..... 18
    - 4.1.3 Service Operation ..... 18
    - 4.1.4 Service Mock-ups ..... 20
  - 4.2 Geofencing Service ..... 21
    - 4.2.1 Service Configuration ..... 22
    - 4.2.2 Service Operation ..... 22
  - 4.3 Physical Training Service ..... 26
    - 4.3.1 Service Overview ..... 26
    - 4.3.2 Service Configuration ..... 27
    - 4.3.3 Service Operation ..... 28
    - 4.3.4 Service Mock-ups ..... 29
  - 4.4 Transnational Journey Planning Service..... 31
    - 4.4.1 Service Overview ..... 31
    - 4.4.2 Service Configuration ..... 32
    - 4.4.3 Service Operation ..... 32
    - 4.4.4 Service Mock-ups ..... 33
  - 4.5 Orientation and Mobility Training Service ..... 35
    - 4.5.1 Hiking Application..... 35
      - 4.5.1.1 Service Overview ..... 35
      - 4.5.1.2 Service Configuration ..... 36
      - 4.5.1.3 Service Operation ..... 36
      - 4.5.1.4 Service Mock-ups ..... 37

4.5.2	Orientation and Geotag Application .....	38
4.5.2.1	Service Overview .....	38
4.5.2.2	Service Configuration .....	40
4.5.2.3	Service Operation .....	40
4.5.3	Indoor / Outdoor Orientation Training Application .....	42
4.5.3.1	Service Overview .....	42
4.5.3.2	Service Configuration .....	43
4.5.3.3	Service Operation .....	43
4.5.3.4	Service Mock-ups .....	44
4.5.4	First Aid Application .....	45
4.5.4.1	Service Overview .....	45
4.5.4.2	Service Configuration .....	46
4.5.4.3	Service Operation .....	46
4.5.5	Checklist Application .....	48
4.5.5.1	Service Overview .....	48
4.5.5.2	Service Configuration .....	49
4.5.5.3	Service Operation .....	49
4.5.6	Orientation and Mobility Games.....	51
4.5.6.1	Service Overview .....	51
4.5.6.2	Service Configuration .....	52
4.5.6.3	Service Operation .....	52
5.	Supported End-User terminals .....	55
5.1	TV Platforms .....	55
5.2	Smart Phone Platforms .....	56
5.3	STBs .....	56
5.4	Tablets .....	56
6.	SW technologies selection .....	58
6.1	Human Machine Interfaces Technologies.....	58
6.2	Communication Technologies .....	59
6.3	Database Technology .....	59
6.4	Web Service Technology .....	59
7.	MOBILE.OLD Application Server Architecture and Smart Phone Architecture.....	62
7.1	Application Server Architecture Overview .....	62
7.2	Smart Phone Software Architecture .....	64

- 8. Development Plan ..... 67
- 9. Evaluation and Deployment Plan ..... 69
  - 9.1 Services Evaluation Plan ..... 69
    - 9.1.1 Mock-Up Evaluation ..... 70
    - 9.1.2 Pre Prototype Evaluation ..... 71
    - 9.1.3 First Prototype Evaluation ..... 71
    - 9.1.4 Second Prototype Evaluation ..... 72
  - 9.2 Deployment Plan ..... 72
    - 9.2.1 User Selection ..... 72
    - 9.2.2 User Training ..... 73
    - 9.2.3 Equipment Acquirement ..... 73
      - 9.2.3.1 Equipment needed: ..... 73
    - 9.2.4 Contract for the Usage of the Equipment ..... 75
    - 9.2.5 Helpdesk Establishment ..... 80
    - 9.2.6 Installation of the MOBILE.OLD Prototypes ..... 83
      - 9.2.6.1 Centrally in the premises of every pilot site ..... 83
      - 9.2.6.2 Installation at user’s households ..... 85
- 10. Services Validation Plan ..... 86
  - 10.1 Personalized Transportation News Headlines Service ..... 86
    - 10.1.1 Tests Configuration ..... 86
    - 10.1.2 Validation Scenarios ..... 86
  - 10.2 Geofencing service ..... 88
    - 10.2.1 Tests Configuration ..... 88
    - 10.2.2 Validation Scenarios ..... 89
  - 10.3 Physical Training Service ..... 92
    - 10.3.1 Tests Configuration ..... 92
    - 10.3.2 Validation Scenarios ..... 92
  - 10.4 Transnational Journey Planning Service ..... 95
    - 10.4.1 Tests Configuration ..... 95
    - 10.4.2 Validation Scenarios ..... 95
  - 10.5 Orientation and Mobility Training Service ..... 97
    - 10.5.1 Hiking Application ..... 97
      - 10.5.1.1 Tests Configuration ..... 97
      - 10.5.1.2 Validation Scenarios ..... 97

10.5.2	Orientation and Geotag Application .....	99
10.5.2.1	Tests Configuration.....	99
10.5.2.2	Validation Scenarios .....	99
10.5.3	Indoor / Outdoor Orientation Training Application .....	102
10.5.3.1	Tests Configuration.....	102
10.5.3.2	Validation Scenarios .....	102
10.5.4	First Aid Application .....	103
10.5.4.1	Tests Configuration.....	103
10.5.4.2	Validation Scenarios .....	103
10.5.5	Checklist Application .....	105
10.5.5.1	Tests Configuration.....	105
10.5.5.2	Validation Scenarios .....	105
10.5.6	Orientation and Mobility Games.....	109
10.5.6.1	Tests Configuration.....	109
10.5.6.2	Validation Scenarios .....	109
11.	Conclusions.....	113
12.	Requirements Coverage Matrix .....	114
13.	Annex A: Web Service Technology Evaluation .....	118
13.1	Introduction.....	118
13.1.1	Methodology .....	118
13.1.2	Requirement Outline.....	119
13.2	Technology Stacks .....	119
13.2.1	JAX-RS .....	119
13.2.1.1	Characteristics .....	119
13.2.1.2	Security .....	120
13.2.1.3	Prototype.....	120
13.2.2	Spring MVC.....	121
13.2.2.1	Characteristics .....	121
13.2.2.2	Security .....	122
13.2.2.3	Prototype.....	122
13.2.3	Django-Piston .....	123
13.2.3.1	Characteristics .....	123
13.2.3.2	Security .....	124
13.2.3.3	Prototype.....	124

13.2.4	Disregarded Technology Stacks.....	125
13.3	Comparison .....	125
13.3.1	Performance Tests.....	125
13.3.1.1	Procedure .....	125
13.3.1.2	Results .....	126
13.3.2	Resource Efficiency.....	126
13.3.3	Difficulty of Development .....	126
13.3.3.1	Measurement Through Lines of Code .....	127
13.3.3.2	Measurement Through Dependency Count.....	127
13.3.3.3	Measurement Through Learnability.....	128
13.4	Summary and Conclusion .....	129

---

## Glossary

---

<b>AAL</b>	Ambient Assisted Living
<b>ADSL</b>	Asynchronous Digital Subscriber Line
<b>AJAX</b>	Asynchronous JavaScript and XML
<b>API</b>	Application Programming Interface
<b>ASQ</b>	After Scenario Questionnaire
<b>CPU</b>	Central Processing Unit
<b>DAL</b>	Data Access Layer
<b>DAO</b>	Data Access Object
<b>GPS</b>	Global Positioning System
<b>GUI</b>	Graphical User Interface
<b>HS</b>	High Definition
<b>HDMI</b>	High Definition Multimedia Interface
<b>HMI</b>	Human Machine Interface
<b>HTTP</b>	Hypertext Transport Protocol
<b>ICT</b>	Information and Communications Technology
<b>ID</b>	Identification
<b>JPA</b>	Java Persistence API
<b>JSON</b>	JavaScript Object Notation
<b>LAN</b>	Local Area Network
<b>MMS</b>	Multimedia Messaging Service
<b>MVC</b>	Model View Controller
<b>OMT</b>	Orientation and Mobility Training
<b>OTT</b>	Over the Top
<b>PC</b>	Personal Computer
<b>POI</b>	Point of Interest

<b>PT</b>	Physical Training
<b>PTNH</b>	Personalized Transportation News Headlines
<b>RAM</b>	Random Access Memory
<b>RDBMS</b>	Relational Database Management System
<b>REST</b>	Representational State Transfer
<b>RSS</b>	Really Simple Syndication
<b>SMS</b>	Short Message Service
<b>SOAP</b>	Simple Object Access Protocol
<b>SQL</b>	Structured Query Language
<b>STB</b>	Set Top Box
<b>SW</b>	Software
<b>TJP</b>	Transnational Journey Planning
<b>TTS</b>	Text-To-Speech
<b>TV</b>	Television
<b>UI</b>	User Interface
<b>URL</b>	Uniform Resource Locator
<b>USB</b>	Universal Serial Bus
<b>WLAN</b>	Wireless Local Area Network
<b>XML</b>	Extensible Markup Language
<b>XSLT</b>	Extensible Style sheet Language

## 1. INTRODUCTION

The MOBILE.OLD project aims to provide a combined Smart Phone and TV-based service infrastructure with residential and outdoor services that will be delivered in a highly personalized and intuitive way and will advance the mobility of older persons. The services will be provided in a cost-effective way aiming at independent living and sustained mobility, reinforcing activation, maintaining the health status, providing cognitive benefits, thus preventing isolation, depression, increased morbidity and loneliness.

MOBILE.OLD will deliver an open platform and services advancing the mobility of older persons and within the project the focus will be on the development of the following four main categories of services:

- **Informational services for assisting mobility**, i.e. services that filter out, either “passively” or “actively”, existing information related to mobility. A “personalized transportation news headlines” service will be provided under this category, which will allow easy access to transportation news headlines at regional, national and European level. The service will allow the older persons to be informed about potential strikes in public transportation, trains, airports, etc. or constructions works that could possibly cause problems to several transportation means or even a visit of politician at a specific place that could introduce constraints in transportation.
- **Travel and transport services**, i.e. services enhancing trans-national mobility. A “trans-national journey planning” service will be provided under this category, which will provide information on how a specific destination can be reached, given a specific location. Such information shall include the different possibilities for transportation means, timetables, guidelines in changing transportation means, ticket issuing etc. The result shall be provided to the user in printable format or by other means depending on the size of information (e.g. SMS).
- **Services for cognitive impaired people**, i.e. services assisting the mobility of older persons suffering from cognitive impairments. A “geofencing” service will be provided under this service category, allowing automatic alerts or notifications when older persons’ entering, leaving or moving within a specific geographic area, thus allowing the detection of being lost and appropriate intervention.
- **Physical and mobility training services**, i.e. services maintaining good health status as well as cognitive games assisting in overcoming situations that older persons may encounter when being mobile. Two services will be provided under this category:
  - A “physical training” service consisting of a virtual trainer that proposes to the older persons several physical exercises. Physical exercise is essential for maintaining good blood flow as well as to encourage new brain cells. The physical exercises will be provided to the older persons at a personalized manner, based on their physical status and planned activities for transportation/travelling in the near future.

- An “orientation and mobility training” service consisting of a set of playful exercises, which shall train the older persons to overcome situations that may encounter when being mobile.

## 1.1 SCOPE

The scope of the current deliverable is to approach the MOBILE.OLD platform from a technical perspective, and gather and analyse the requirements that will lead to and guide through its implementation. Specifically, the scope of the current deliverable is:

- To select and specify in detail the services to be implemented, deployed and piloted.
- To select the equipment required for the implementation and deployment of the services.
- To select the SW technologies for the implementation of services application logic, HMIs and communication interfaces.
- To specify the MOBILE.OLD network architecture and its components.
- To define the development plan and test and validation plan for the MOBILE.OLD services.

## 1.2 DELIVERABLE STRUCTURE

The current deliverable is composed of twelve chapters.

- Chapter one provides the introduction to the document, and undertakes the analysis of its scope.
- Chapter two undertakes the analysis of the MOBILE.OLD conceptual architecture, providing an overview of its components.
- Chapter three describes the methodology and the consideration that were taken for the selection of the MOBILE.OLD services that will be implemented.
- Chapter four provides the high level specifications of the MOBILE.OLD services which include the Personalized Transportation News Headlines service, the Geofencing service, the Physical Training service, the Transnational Journey Planning service and the Orientation and Mobility Training service.
- Chapter five provides a description of the end-user terminals which include Smart Phone and TV-based platforms, Tablets and over the top STBs.
- Chapter six provides a technical description of the HMI, communication and database software technologies that will be used to implement the MOBILE.OLD services.
- Chapter seven undertakes the technical analysis of the MOBILE.OLD architecture. The chapter includes the architectural description of the MOBILE.OLD Application Server along with the description of its modules as well as the architectural description of the Smart Phone SW architecture.
- Chapter eight provides the MOBILE.OLD software development plan, including the software implementation and integration of each system component.

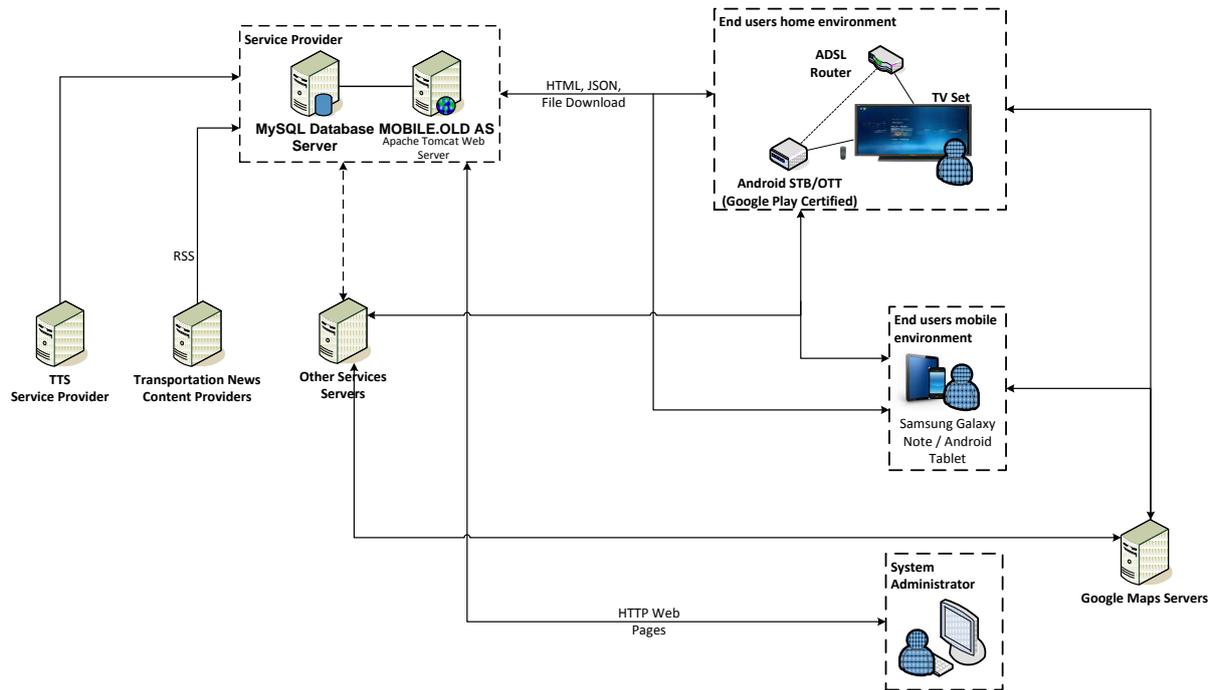
- Chapter nine describes the evaluation and deployment plan of MOBILE.OLD, from the perspective of the end-user trials.
- Chapter ten provides the technical validation plan of the MOBILE.OLD services.
- Chapter eleven provides the conclusions to this deliverable.
- Chapter twelve provides a requirements traceability matrix depicting the section of this document that each requirement selected in the context of D2.1 is addressed.

### **1.3 METHODOLOGY**

Based on the user requirements and use cases defined in D2.1, the MOBILE.OLD services were specified in detail. Before specifying the MOBILE.OLD architecture, it was necessary to identify and select the platforms of the end-user endpoints. With this background information, the MOBILE.OLD architecture was defined and specified, along with the software technologies that will be used for its implementation. Finally, the evaluation and deployment plan for the user trials was defined, along with the technical validation plan of the MOBILE.OLD services.

## 2. MOBILE.OLD NETWORK ARCHITECTURE

The following figure provides an overview of the MOBILE.OLD network architecture.

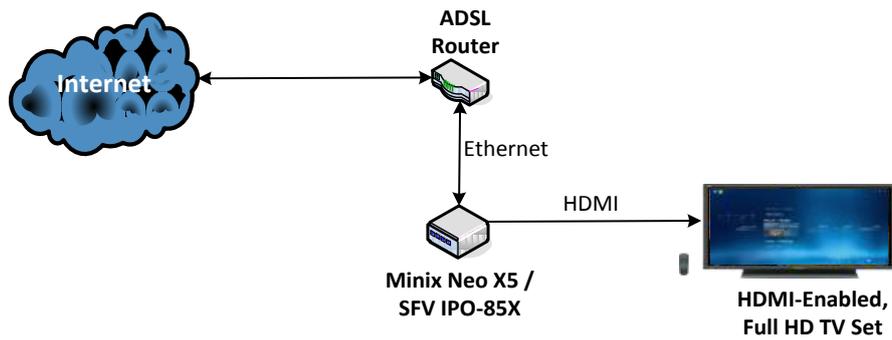


**Figure 1: MOBILE.OLD Network Architecture**

The MOBILE.OLD network architecture consists of the following sites:

- **Service provider**, where the MOBILE.OLD Application server and services data reside.
- **MOBILE.OLD Other Services Servers**, which contains the MOBILE.OLD servers that implement and support the Geofencing and Orientation and Mobility Training services.
- **Transportation News Content Providers** that provide content to the Personalized Transportation News Headlines.
- **TTS Service Provider**, that provides the TTS functionality of the services.
- **Google Maps Server**, that provides the mapping functionality to the Transnational Journey Planning and Indoor / Outdoor Orientation Training services.
- **End users home environment**, which access the MOBILE.OLD services using a TV based platform accessing the platform through Android STB/OTT applications.
- **End users mobile environment**, who access the MOBILE.OLD services using the Smart Phone and tablet endpoints.
- **System Administrator**, who connect to the MOBILE.OLD services to manage the services and provide support to the end users.

The user's home environment network architecture is depicted in the following figure.



**Figure 2: MOBILE.OLD Home Environment Network Architecture**

The home environment is composed of the following components:

- **HDMI-Enabled Full HD TV Set**, which is used as a monitor to display the services applications.
- **Minix Neo X5 OTT or SFV IPO-85X STB**, which are Android devices running the services applications and connect to the Internet via the ADSL router.
- **ADSL Router**, which provides Internet connectivity to the Android STB/OTT.

### 3. SELECTION OF THE SERVICES TO BE IMPLEMENTED

This section provides the set of the services that are planned to be developed in the context of the MOBILE.OLD project. The main selection criteria include the rating of the different services that has been performed by the End Users during the requirements collection phase as well as market trends that would allow the successful commercialization of the MOBILE.OLD services after the end of the project. Towards this direction, the following priorities have been considered:

- Priority “1”: Services considered to be implemented in very short term (immediately and during the project course).
- Priority “2”: Services considered to be implemented in short to medium term (after the end of the project).
- Priority “3”: Services considered to be implemented in medium to long term (after 3 years counting from the end of the project).

The following table provides an overview of the MOBILE.OLD services and applications, their rating by the End Users as well as their prioritization for their implementation.

Service type	Interested (Very + Fairly)	Not Interested	Priority for service implementation
Personalized Transportation News Headlines	78,8%	21,2%	1
Weather forecast App	75,8%	24,2%	3
Geofencing incl. Detection of External Physical Conditions	75,8%	24,2%	1
First Aid App	72,7%	27,3%	1
Physical Training	66,7%	33,3%	1
Checklist App	63,6%	36,4%	1
Orientation App	63,6%	36,4%	1
Geotag App	60,6%	39,4%	2
Transnational Journey Planning	57,6%	42,4%	1
Traffic Sign Memory and Quiz	57,6%	42,4%	1
Hiking App	54,5%	45,5%	1
Virtual Ticket Printer App	54,5%	45,5%	3

**Figure 2: Consolidated services importance per user group and services prioritization for implementation**

Although the “Weather forecast App” is highly rated, it was decided not to be implemented as a standalone application due to the existence of many similar applications on the market. Instead, such an application feature (weather forecast) is considered to be integrated with other MOBILE.OLD services such as the personalized transportation news headlines service.

Additionally, the “Geotag app” was decided to be combined with the “Orientation App” since the applications’ features are directly linked together and they can provide a really useful and competitive application for the AAL market.

## 4. MOBILE.OLD SERVICES SPECIFICATION

### 4.1 PERSONALIZED TRANSPORTATION NEWS HEADLINES SERVICE

#### 4.1.1 SERVICE OVERVIEW

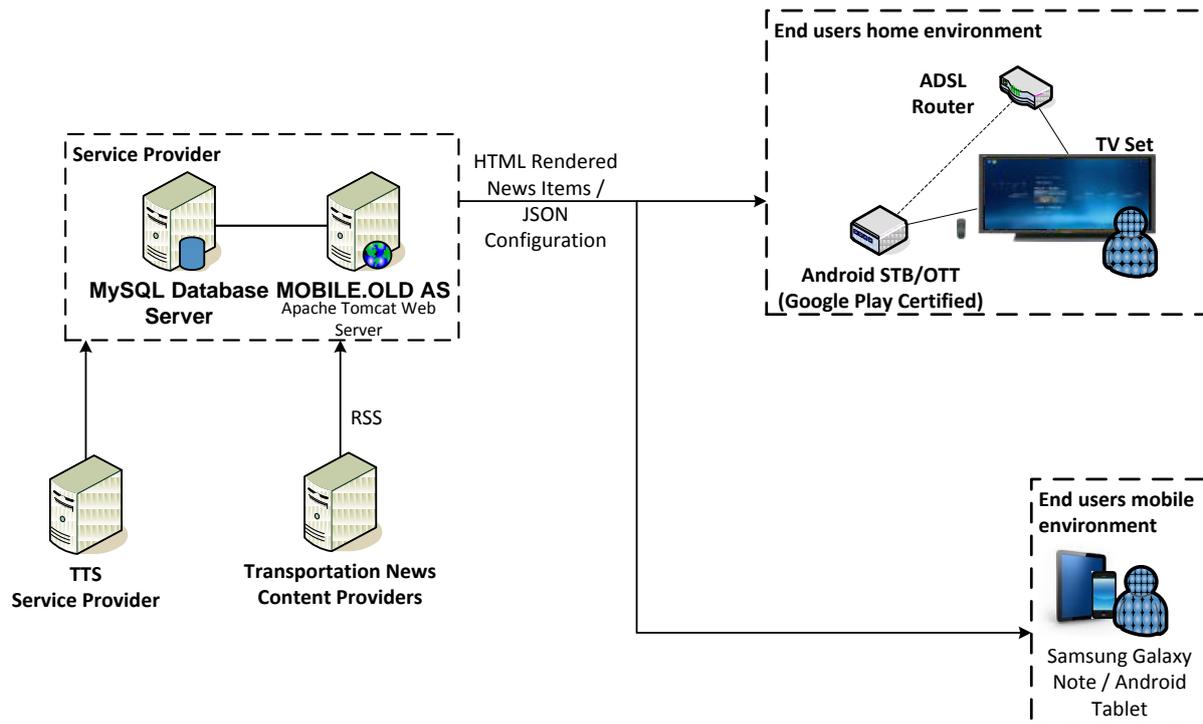
This service provides easy access to transportation news headlines to elder users via a Smart Phone and a TV application. The service offers aggregated content from 3<sup>rd</sup> party sources and provides regional, national and European transportation news with special emphasis on news that correlate with a planned journey or current location of elderly people. Also the service connects to an online weather service and provides weather information on the user's current location or planned journey.

The service receives from the 3rd party news content providers a reliable and fitting list of news headlines, which gets aggregated and filtered according to the elder person's preferences before delivering. The order and way of presentation of the news headlines, as well as the elder person's preference on transportation means, can be made in a highly personalized way based on preferences that are made available by the MOBILE.OLD personalization framework. Also, each news item can be read out loud via a Text-to-Speech system.

Overall features:

- Transportation news from multiple 3rd party content/service providers
- Weather information for the user's current or planned location
- Service personalization menu access
- Filtering of the retrieved news based on user's preferences
- Text-to-Speech support of news headlines
- Link with the Journey Planning service (TJP)
- News on a predefined journey taken from TJP
- TV and mobile devices application

The following figure shows a high level architecture diagram of this service:



**Figure 3: High level architecture of the Personalized Transportation News Headlines**

The elder person uses his Smart Phone, Tablet or TV (when in the home environment) and accesses the MOBILE.OLD service through the internet. The Transport News Headlines are provided by the 3<sup>rd</sup> party content and service providers and the MOBILE.OLD service filters the acquired News Headlines based on elder person's personalization preferences.

#### 4.1.2 SERVICE CONFIGURATION

The service's main configuration is provided through the MOBILE.OLD Personalization Framework, which is also accessible through the service's personalization menu.

The elder person is able to set his/her preferences on Transportation means and the regional scope of the received news headlines, the number of news items per page as well as accessibility preferences such as contrast level, font size, etc.

#### 4.1.3 SERVICE OPERATION

The service is accessible to the user either by using his mobile device or his TV set. The first time the user accesses the service a personalization wizard appears where displaying a menu with two main categories: Area news and Transportation news. The Area news category will contain the Regional, National and European options while the Transportation news category will contain the transportation options. The options in both those menus are multi-selectable, in order for the user to set his exact area and transportation preferences. The option to re-personalize is available at any time.

The service's main screen contains the following items:

- Ground transportation
- Air transportation
- Strikes information
- Traffic information
- Weather information
- Link to TJP service

The transportation news can be regional, national and European. The scope of the transportation news is configurable via the personalization menu.

If a predefined journey has been planned and the user wishes to re-route it or supplement it with more transportation options, he will be able to directly visit the TJP service from the provided link in the service's main menu. For example, a user has planned the initial journey only with bus as transportation and wishes to change his preference to bus and train, so the journey gets re-routed according to these transportation means. The user he visits the TJP service, changes his preferred transportation option and then, back in the Personalized Transportation News Headlines service, he is able to view the news on the new journey preferences.

For a predefined journey, the TJP service provides the areas that are related with that route, so the Personalized Transportation News Headlines retrieves news (and weather information) on the user's destination.

The service, through the 3<sup>rd</sup> party news headlines providers, is able to provide information to the user for eventual transportation irregularities, like for example bad weather on the destination of the journey or strikes on the public transportation systems. In summary, the service will offer the following types of news:

- Weather on the destination
- Public transportation systems strikes and general status
- Traffic information across the entire route

Each displayed news item will be selectable and the user will be able to hear it via the Text-to-Speech functionality, send it to one of his mobile contacts via SMS or email it.

### 4.1.4 SERVICE MOCK-UPS

The following figures depict the user interface mock-ups for the Personalized Transportation News Headlines Service.

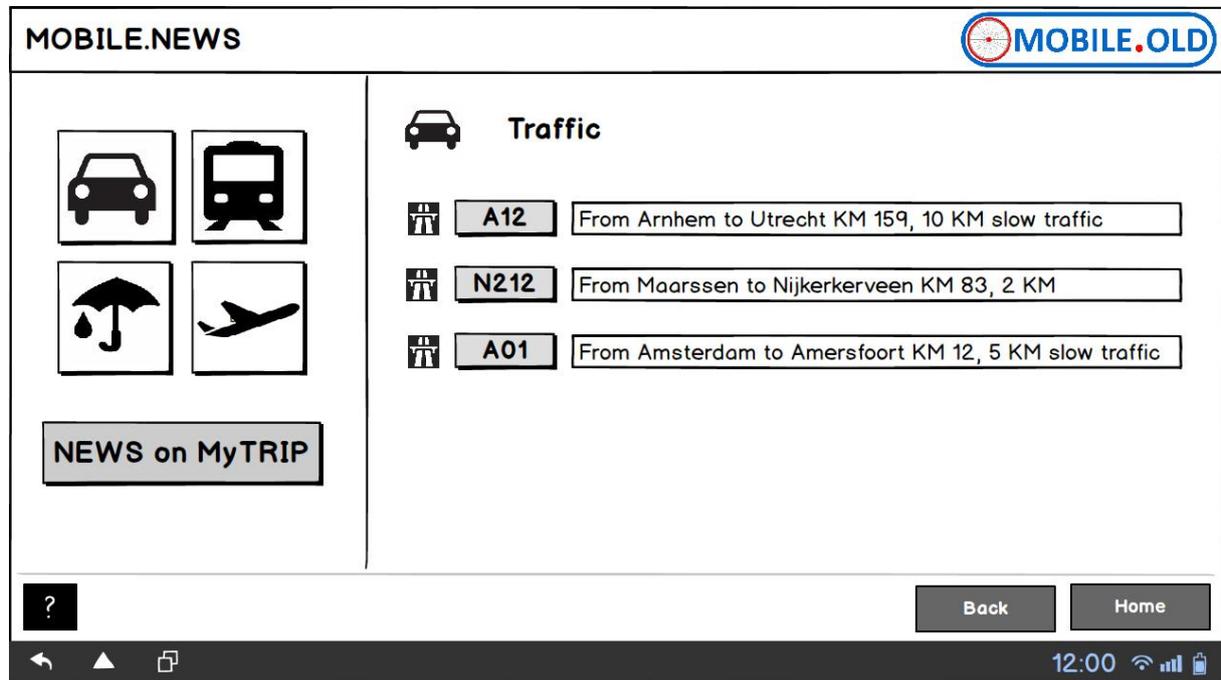


Figure 4: PTNH Service - Traffic News Screen on the TV

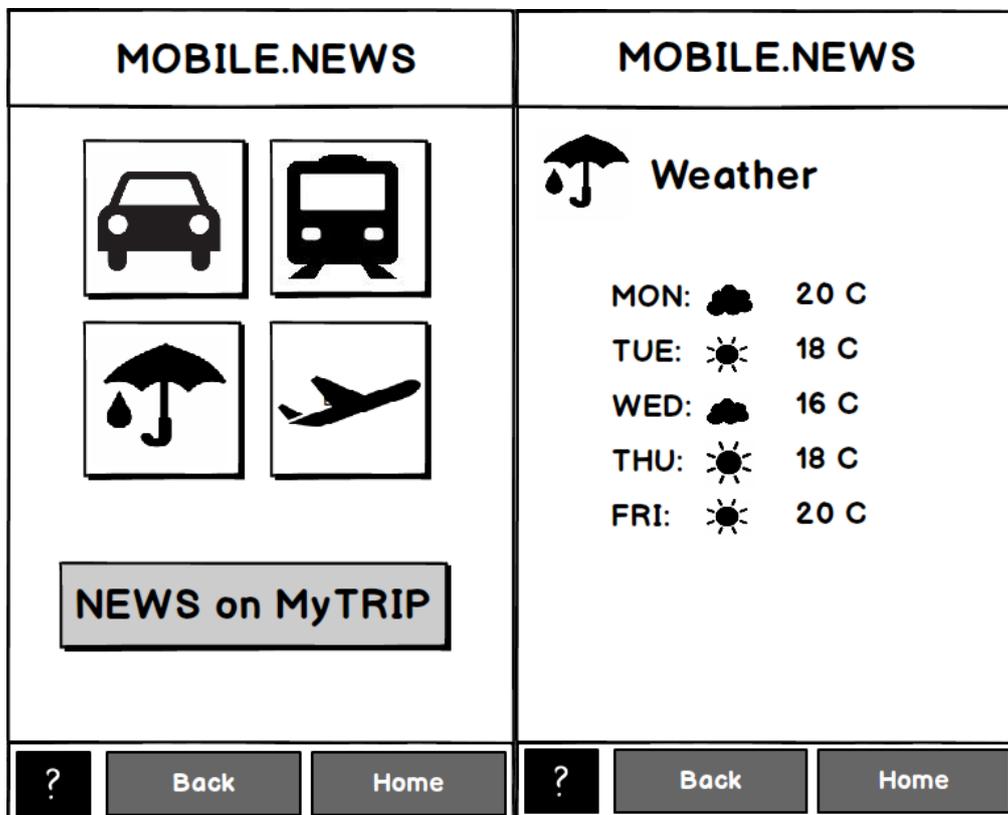
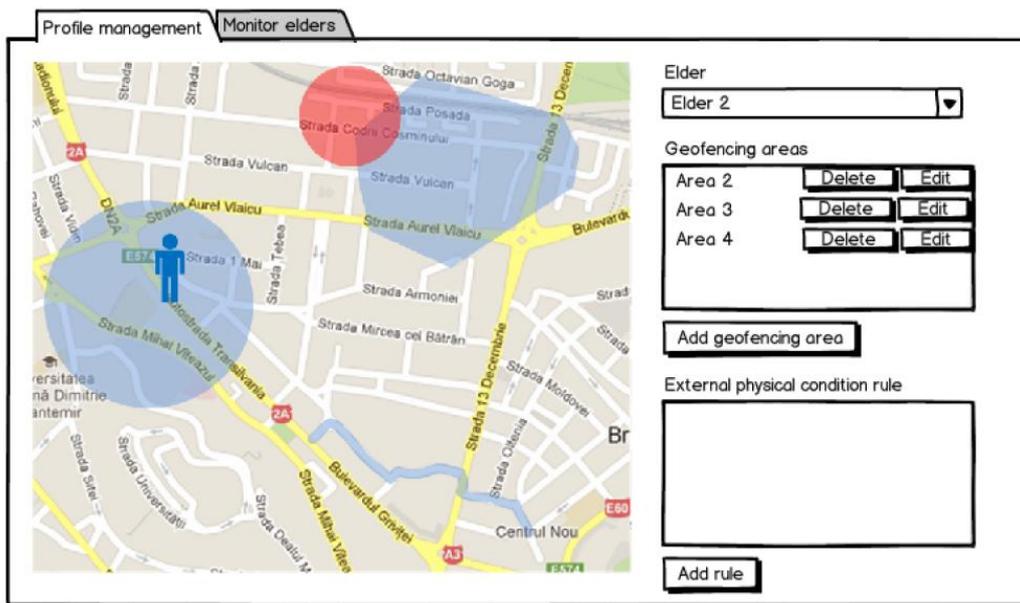


Figure 5: PTNH Service - Home & Weather Screen on the Smart Phone

## 4.2 GEOFENCING SERVICE

The geofencing service provides mobility assistance for persons suffering from cognitive impairments while performing outdoor activities. The locations of the elders are constantly monitored and automatic alerts and notifications are sent when any deviation from the planned route is detected. On the first stage the alert is sent only to the older person or to the older person's family or caregiver if he or she does not return to the geofencing area during a certain time interval (defined by the profile). The geofencing areas can be defined as (see Figure 6):

- an area within a predefined radius from a selected location;
- an envelope around a planned route;
- An area defined by any closed polygonal chain shape.



**Figure 6: Geofencing service mock-up: definition of geofencing areas**

In addition, the service monitors the matching of predefined patterns (sequences of events) which could indicate that the older person has or might have a problem generated by external physical conditions.

The following figure shows the geofencing service architecture diagram.

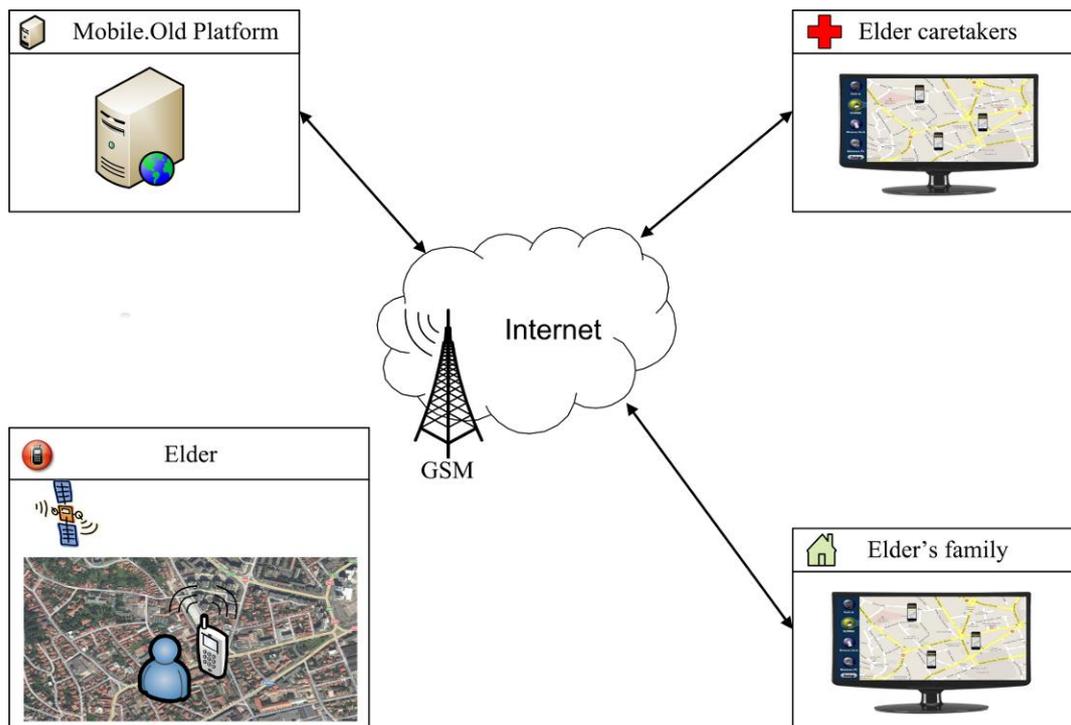


Figure 7: Geofencing service - architecture diagram

#### 4.2.1 SERVICE CONFIGURATION

The service provider, which commercially exploits the MOBILE.OLD platform and its services, usually has the system administrator rights and it is responsible for deploying the application at the client. As a result, the service provider is responsible for deploying the MOBILE.OLD application on the devices (Smart Phones, computers etc.) and for creating user accounts for the users: elders, caretakers, family members.

#### 4.2.2 SERVICE OPERATION

##### Participating Actors:

The **system administrator** creates the user accounts for elders, caretakers and family members.

For the **elder caretakers or the elder's family** the MOBILE.OLD application has two working modes: elders' profile management and elders' monitoring. In the elders' profile management mode the user has the possibility to create/remove/edit the geofencing areas (as presented in Service Overview section) for each elder. The users can also define hazardous areas (within the geofencing area) which will be applied for all the elders registered on the MOBILE.OLD application (e.g. temporary construction works on an intersection, crowded events). Furthermore, in this working mode the elder caretaker or the family can edit the patterns for the detection of external physical conditions (DEPC). The user can use the following patterns (in any combination) where he or she can configure  $X$ ,  $Y$ ,  $t$  parameters (see Figure 8):

- temperature larger than  $X$  °C and older person outside more than  $t$  minutes;
- temperature smaller than  $X$  °C and older person outside more than  $t$  minutes;
- atmospheric pressure larger than  $Y$  mmHg and older person outside more than  $t$  minutes;
- atmospheric pressure smaller than  $Y$  mmHg and older person outside more than  $t$  minutes;
- a variation of temperature larger than  $X$  °C in a period of time smaller than  $t$  minutes;
- a variation of atmospheric pressure larger than  $Y$  mmHg in a period of time smaller than  $t$  minutes.

The screenshot shows a web interface with two tabs: 'Profile management' and 'Monitor elders'. The 'Monitor elders' tab is active. Below the tabs, the text 'Elder 2' is displayed. A 'Rule name' field contains 'Rule 1'. There are four radio button options for defining a rule:

- temperature  than  °C and older person outside more than  minutes. A dropdown menu below 'larger' shows 'smaller' as an alternative.
- atmospheric pressure  than  mmHg and older person outside more than  minutes.
- a variation of temperature larger than  °C in a period of time smaller than  minutes.
- a variation of atmospheric pressure larger than  mmHg in a period of time smaller than  minutes.

At the bottom of the form are two buttons: 'Save' and 'Cancel'.

**Figure 8: Geofencing service mock-up: predefined patterns for external physical condition monitoring.**

In the elders monitoring mode (see Figure 9), the MOBILE.OLD application displays a map where the positions of the elders and the geofencing areas are indicated. In this working mode the elder caretaker or family can use a set of filters for the information displayed on the map (e.g. all elders' positions, only one elder position with a specific ID, elders' positions and geofencing areas, only elders positions outside the geofencing areas, etc.).

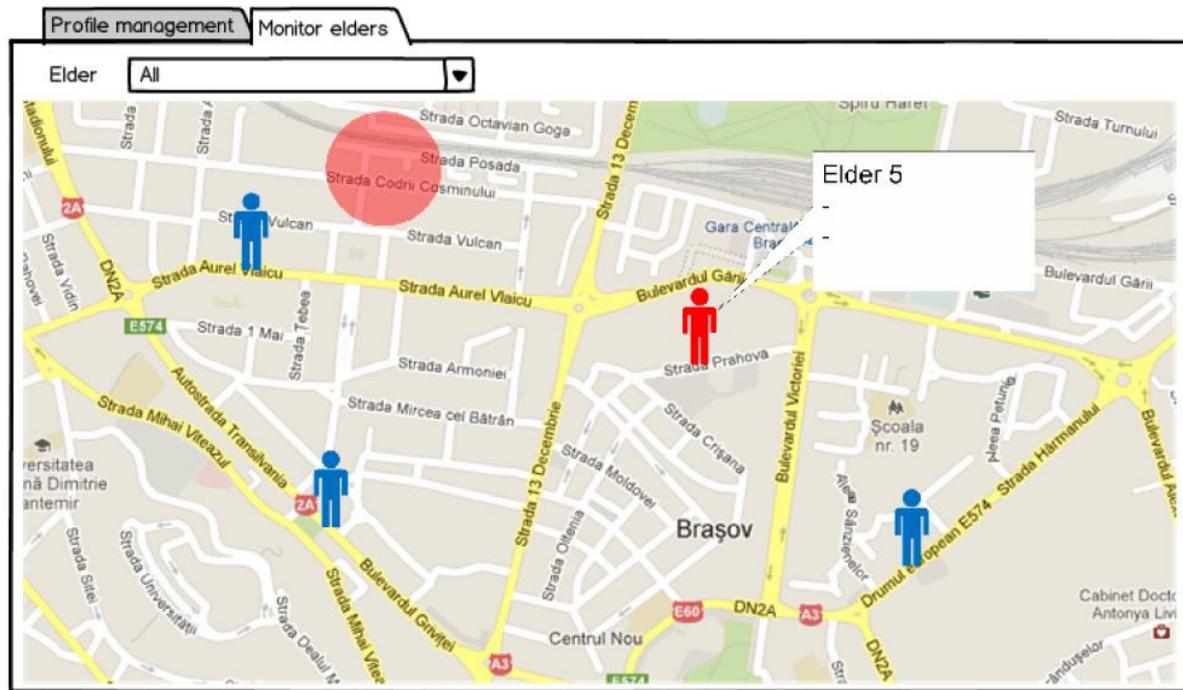


Figure 9: Geofencing service mock-up: elders monitoring mode.

The **elder caretakers or the family** will receive a SMS message in the following situations:

- The elder is outside the geofencing area and does not acknowledge the notification message;
- The elder is outside the geofencing area, he or she acknowledges the notification message but does not return to the geofencing area within a certain time interval specified by the elder profile;
- An external physical condition is met.

The **elderly user** carries a Smart Phone having the MOBILE.OLD application running with the geofencing areas defined by his or her caretaker or family members. When the service detects the elder being outside the geofencing area it generates a warning on the older person's Smart Phone (the phone starts to ring or/and vibrate and a notification message is displayed on the screen). The elder should check the notification message and acknowledge the warning by pressing a button on the Smart Phone (see Figure 10).

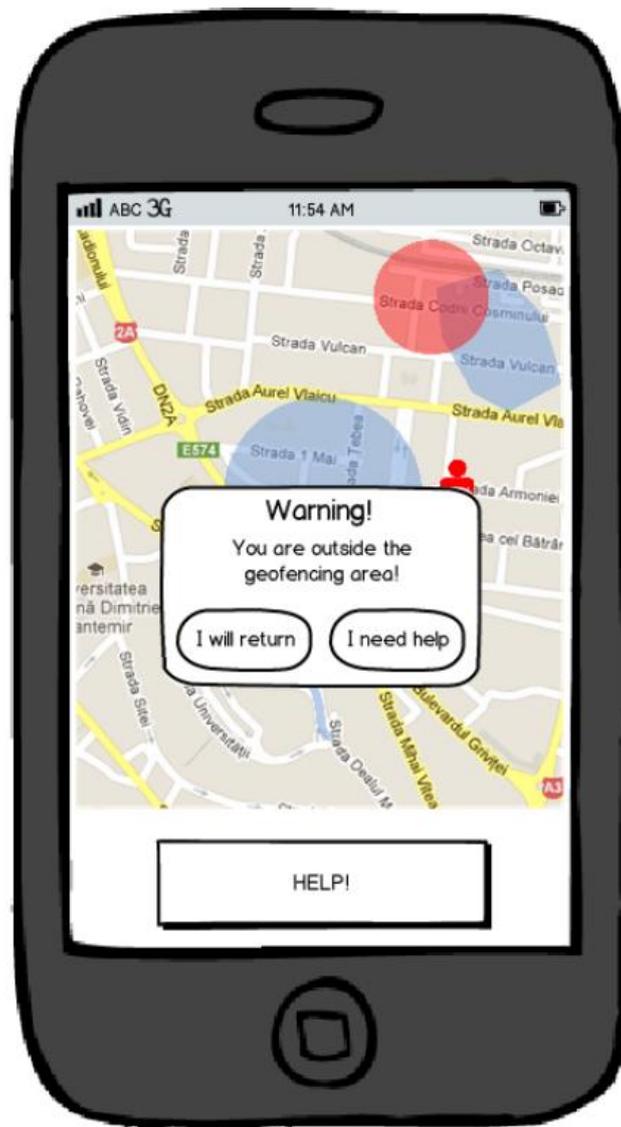


Figure 10: Geofencing service mock-up: the warning generated by the Smart Phone when the elder gets outside from a geofencing area.

The geofencing service workflow is presented in the following diagram.

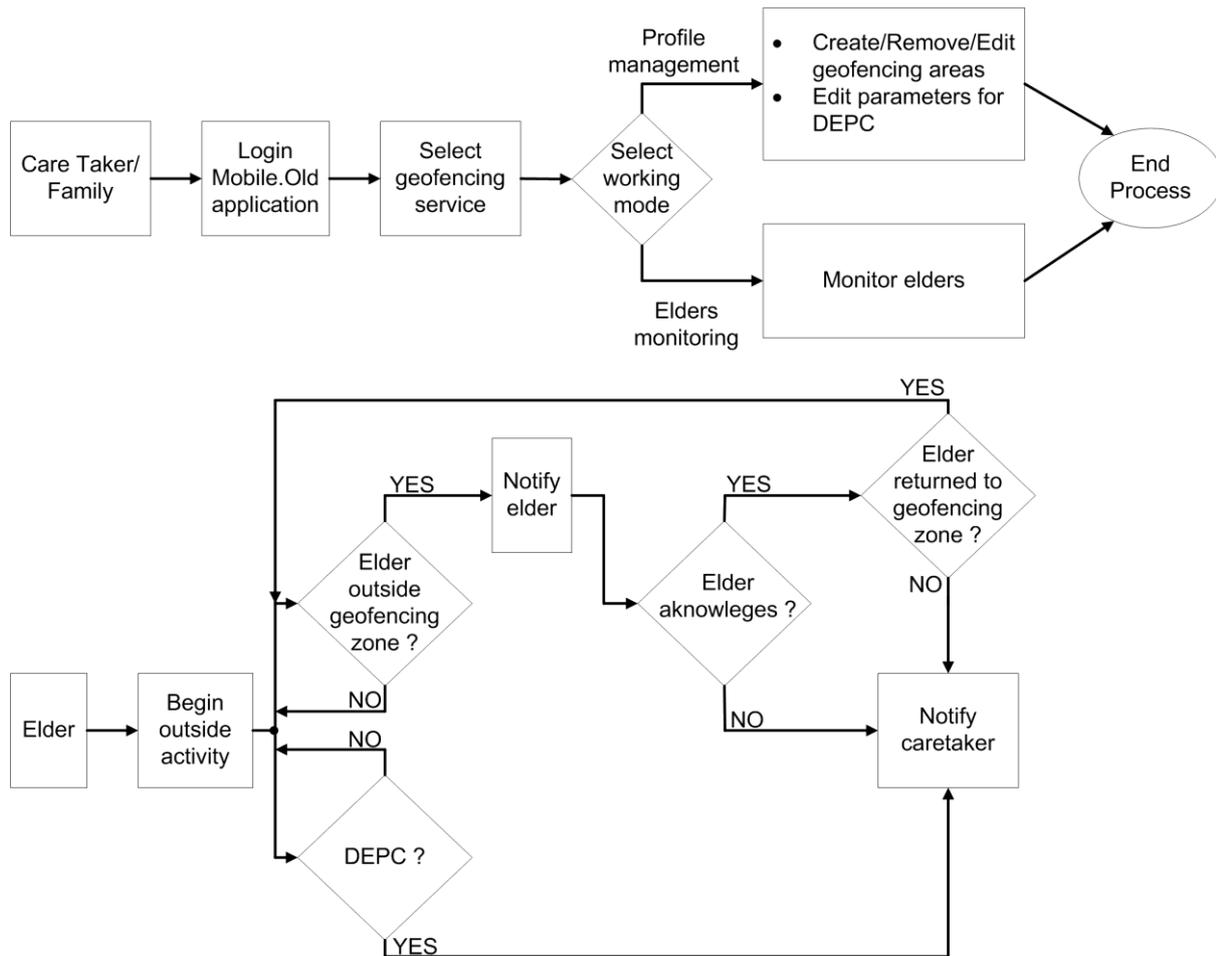


Figure 11: Geofencing service workflow

### 4.3 PHYSICAL TRAINING SERVICE

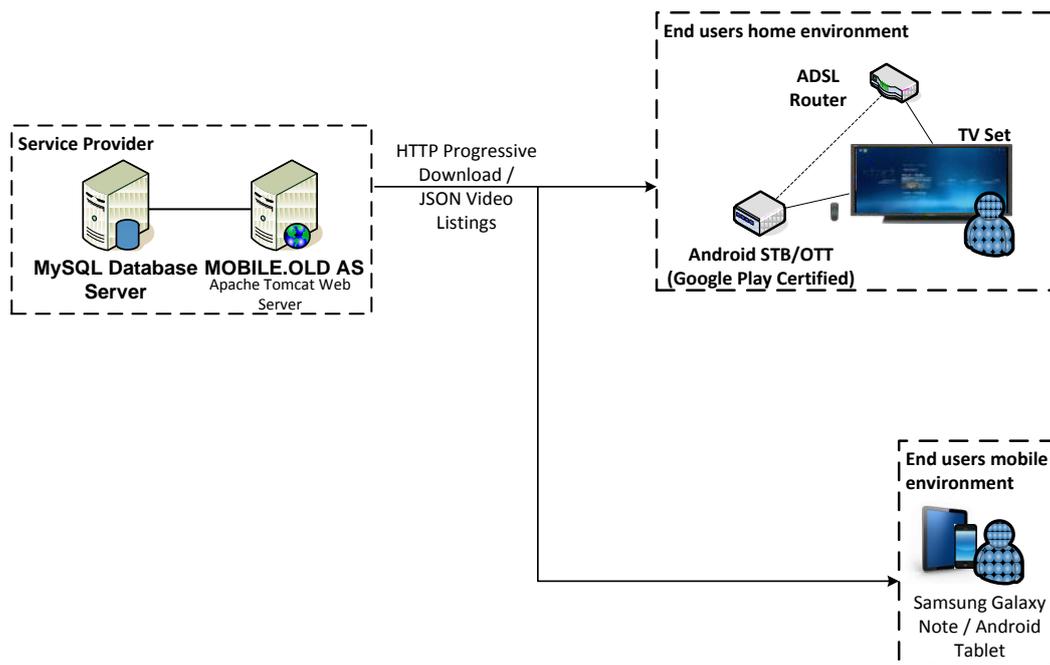
#### 4.3.1 SERVICE OVERVIEW

The Physical Training service provides various proposed multimedia exercises to the users, allowing them to enhance their physical status and thus lead a healthier life. The exercises are provided on demand in the form of video tutorials and the user has the ability to start/stop their playback. The video exercises are categorized based on scope and the user can schedule them to watch them at a later time.

The service is accessible either via TV or mobile device. It provides physical exercise reminders based on user preferences and includes an emergency alarm functionality, which is used in case the user faces problems during the exercises so that a relative or a caregiver can get notified. The exercise videos are reviewed by a doctor so that they are deemed safe for elder individuals and are uploaded to the service by the system administrator.

Users can also update their physical status (e.g. poor, moderate, etc.) in the MOBILE.OLD personalization menu before or after taking an exercise. Also, they can alter other personalization options that are provided, such as contrast level, font size, etc.

The following figure shows a high level architecture diagram of this service:



**Figure 12: Physical Training high level architecture diagram**

The elder person uses his mobile device or TV (when in his home) and contacts the MOBILE.OLD service through the Internet. The TV is used for enabling the elderly to view the exercise videos. . The mobile device covers the same features as the TV application except the exercise video play back. Additionally the reminders are only available on the mobile device.

Overall features:

- TV and mobile application
- Exercise videos on demand (on the TV)
- Scheduling and reminding of exercise videos
- Alarm button to alert in case of emergency
- Personalization framework/menu access

#### 4.3.2 SERVICE CONFIGURATION

The service's main configuration is provided through the MOBILE.OLD Personalization Framework, which is also accessible through the service's Personalization menu item. The elder person is able to set his ability to stand and/or his physical status to poor/moderate/good/excellent as well as accessibility preferences such as contrast level, font size, etc. Also, the user is able to schedule the video exercises that he wants to watch at a later time and be reminded.

The MOBILE.OLD application server features a backend for the Physical Training service. Through that backend, the system administrator can upload new exercise videos and manage their categories.

### 4.3.3 SERVICE OPERATION

The main screen of the application displays a list of pre-defined categories each of which contains a set of video exercises. The user can view information about each of the exercises such as type (endurance, strength, balance, flexibility) and duration that are displayed together with a start and a stop button for that selection. Also, the user can playback the exercise by pressing the start button or schedule the exercise for the future using the dedicated schedule button. In case the user has problems and wants to stop the exercise before it is finished, he can select the stop button. After completing an exercise the user will get the option to update his physical status if it has been altered.

The physical exercise reminder functionality is available for the exercise videos that the user has selected to view at a later time. A reminder pops up to the user's mobile device reminding him that there is an exercise that the user has scheduled.

The emergency alarm functionality is accessible in all kinds of exercises, which is used in case of an emergency during an exercise in order to alert a relative or a caregiver. A dedicated emergency button is available which when pressed notifies the predefined relative/caregiver.

In summary, the functionality of the service will be provided by the following user interface screens.

GUI Screen	Functionality
Exercises	Provides a list of proposed or scheduled exercises based on the user's personalization preferences.
Personalization	Allows the user to set his preferences for the service.

**Table 1: Physical Training service GUI screens**

### 4.3.4 SERVICE MOCK-UPS

The following figures depict user interface mock-ups of the Physical Training service.

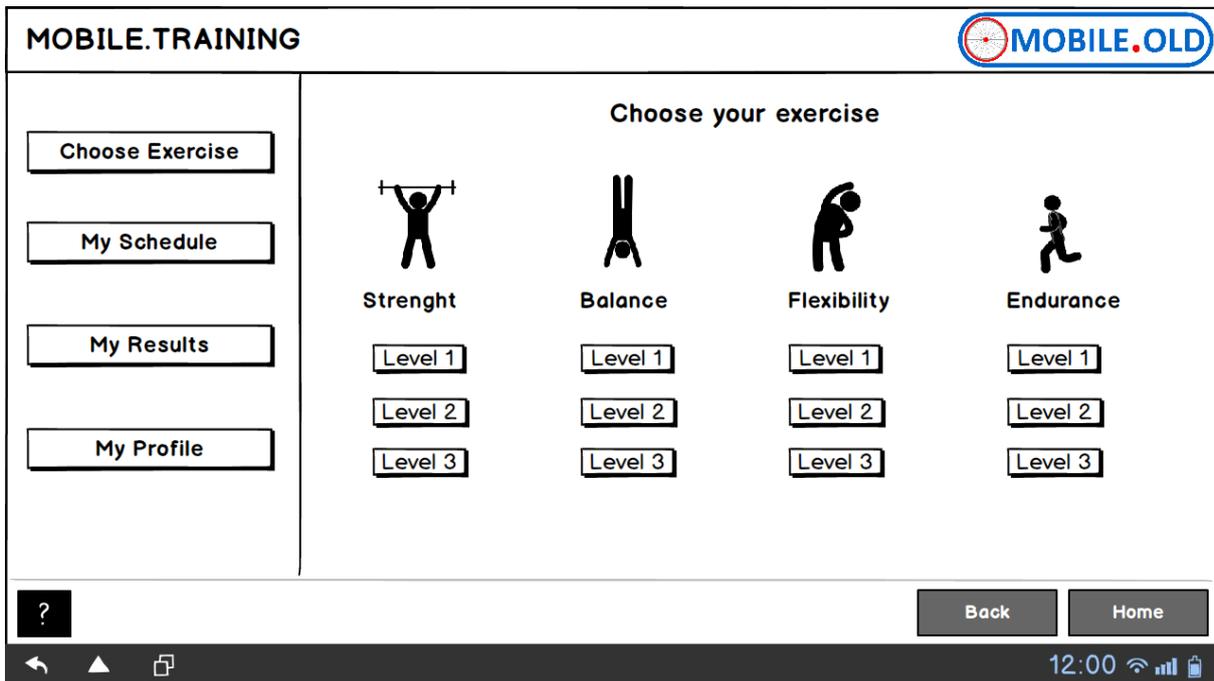


Figure 13: PT Service - Home Screen on the TV

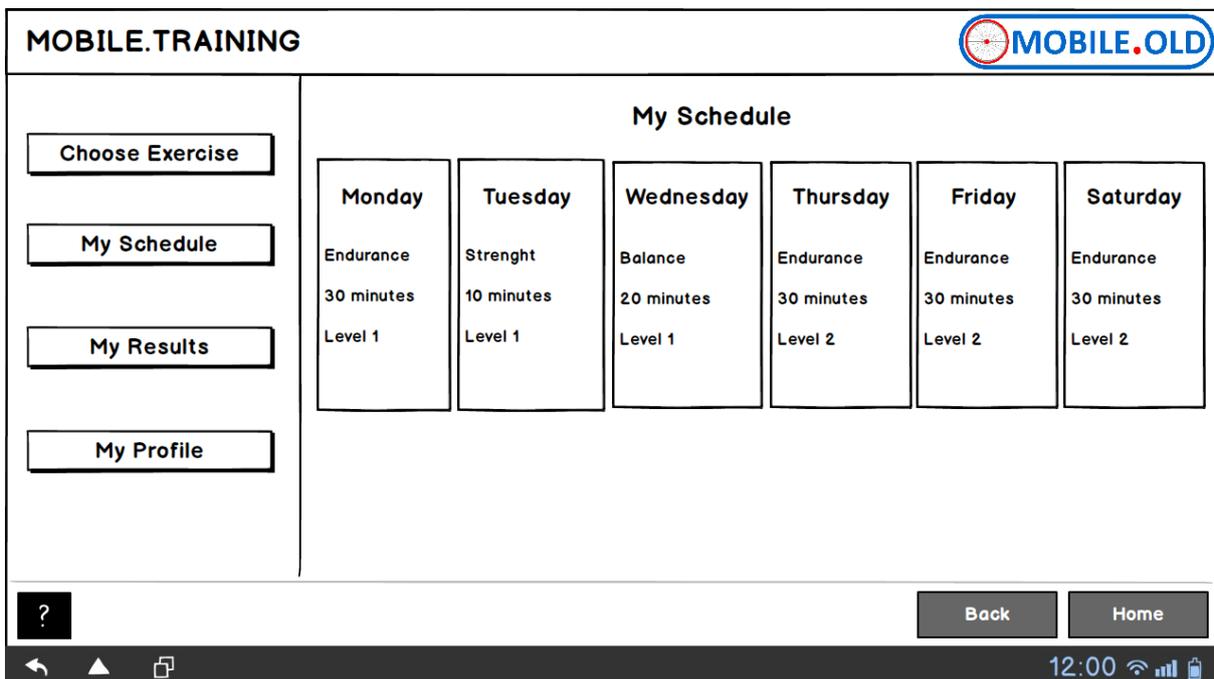


Figure 14: PT Service - Schedule Screen on the TV

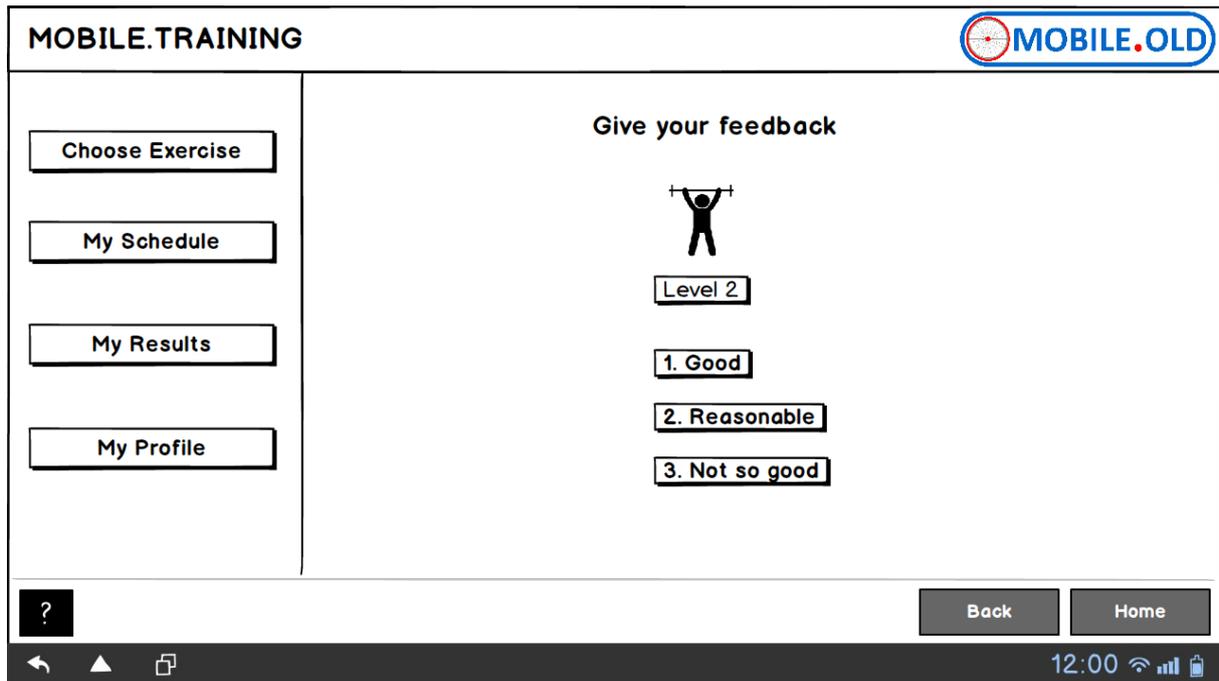


Figure 15: PT Service - Feedback Screen on the TV

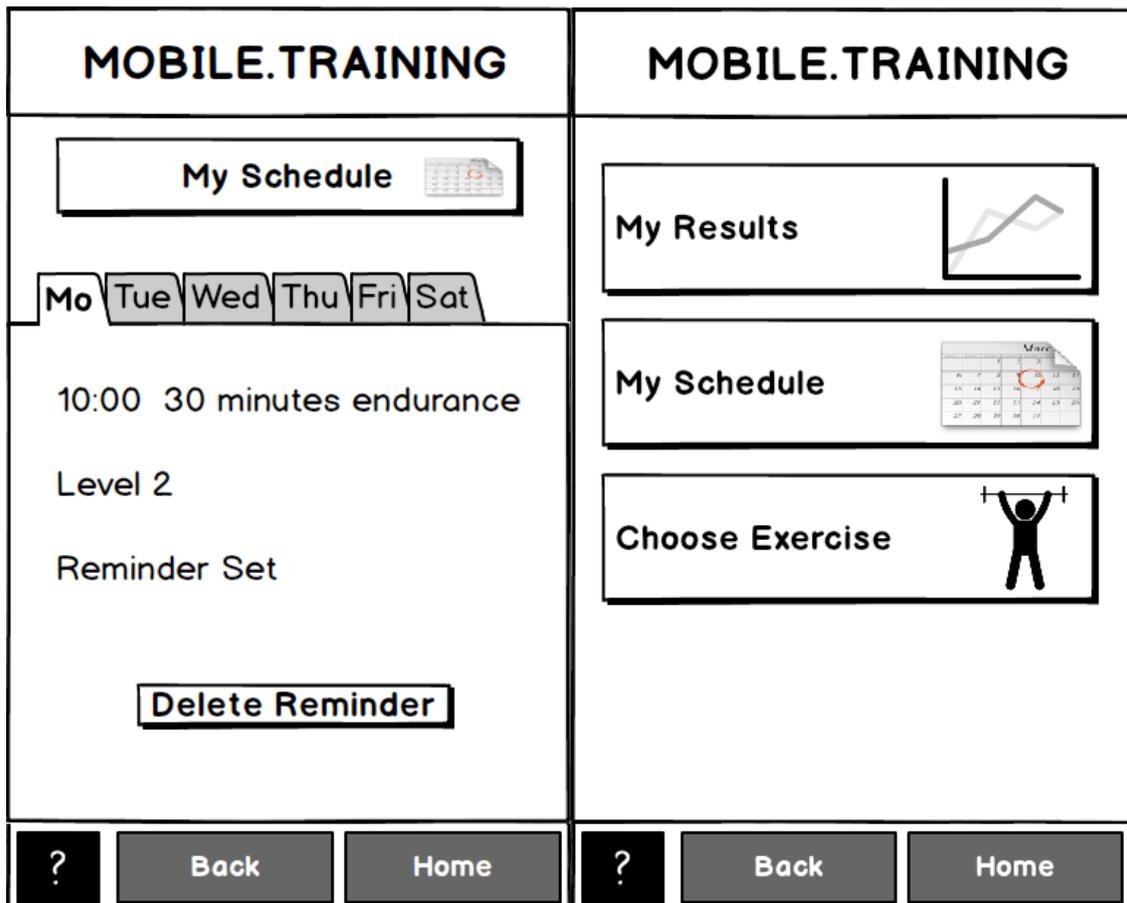


Figure 16: PT Service - Reminder & Home Screen on the Smart Phone

## 4.4 TRANSNATIONAL JOURNEY PLANNING SERVICE

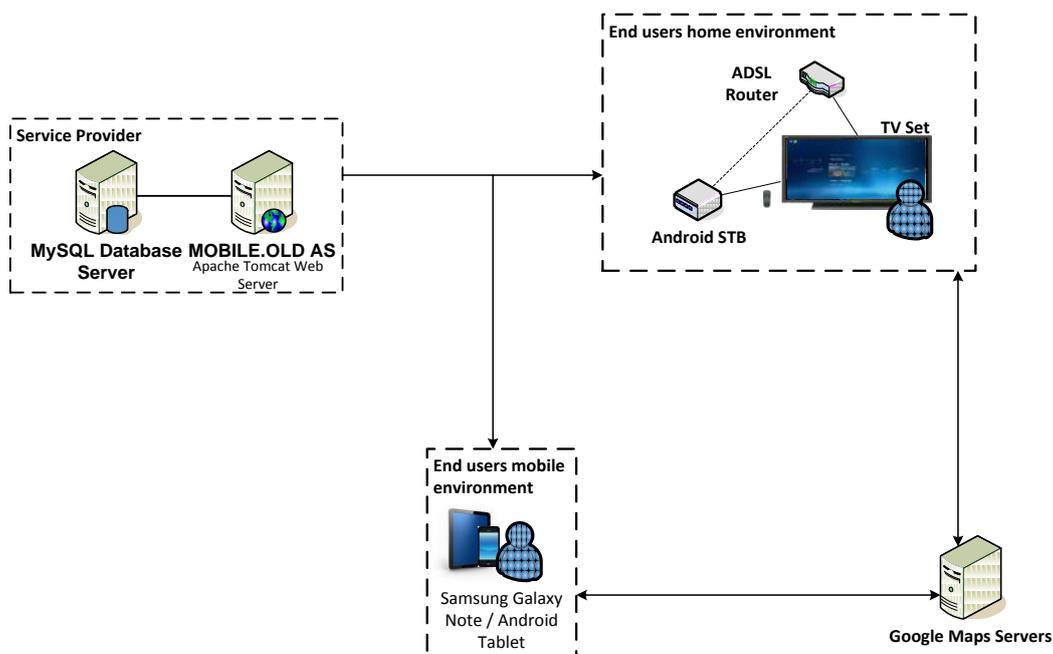
### 4.4.1 SERVICE OVERVIEW

The Transnational Journey Planning service allows the user to select an origin and a destination location, along with a transportation means, and provides information on a recommended and alternative travel routes and travel time.

The service is accessible from both the home and mobile environments of MOBILE.OLD and allows the user to access, configure, calculate and save his trip on whatever endpoint he is more comfortable with.

When planning a trip, the service takes into account the user's mobility level to calculate the most convenient for the user route. Also, the service allows the user to select whether a car, a public transportation or a bike will be used as transportation means for the trip and changes the travel route accordingly. In addition to that, it will also be possible to calculate the route if the elderly wants to go by foot.

The following figure depicts a high level architectural diagram of the service.



**Figure 17: Transnational Journey Planning service high level architecture diagram**

The user uses his TV, when in the home environment or Smart Phone/tablet device when in the mobile environment to access the MOBILE.OLD Application Server which implements the service. The synchronisation between all the endpoints happens automatically.

#### **4.4.2 SERVICE CONFIGURATION**

The service's main configuration is provided through the MOBILE.OLD Personalization Framework, which is also accessible through the service's Personalization menu item. The user is able to set his mobility level so that the service can calculate the most relevant routes. Also, for each trip, the user is able to select on top of the origin and destination, the default transportation method (e.g. car, public transportation, walking, etc.) that he prefers to use for his trip.

#### **4.4.3 SERVICE OPERATION**

The main menu of the service features text boxes for the destination and origin country, city and address, depending on the travel scope (regional, national or transnational). The origin fields are filled with the user's default configuration. Upon selecting one of the textboxes, the user is prompted to type the origin/destination of his trip. Valid inputs to the origin/destination textboxes include cities and specific addresses.

The main menu of the service also features a "Calculate Journey" button which upon pressed, will display the "Route" screen which will display the recommended travel routes from the selected origin to the destination.

When saving a calculate route, the senior will have the possibility to save the calculated route to the hiking service, in order to record and follow the route (track) on the Smart Phone application.

When pressing the send button the elderly user is able to send his route with an e-mail attachment to his friends, relatives or even caregivers. If the elderly user himself receives a route, via e-mail (by opening the attachment) the route is automatically put to the my routes section of the service.

### 4.4.4 SERVICE MOCK-UPS

The following figures depict user interface mock-ups of the Transnational Journey Planning service.

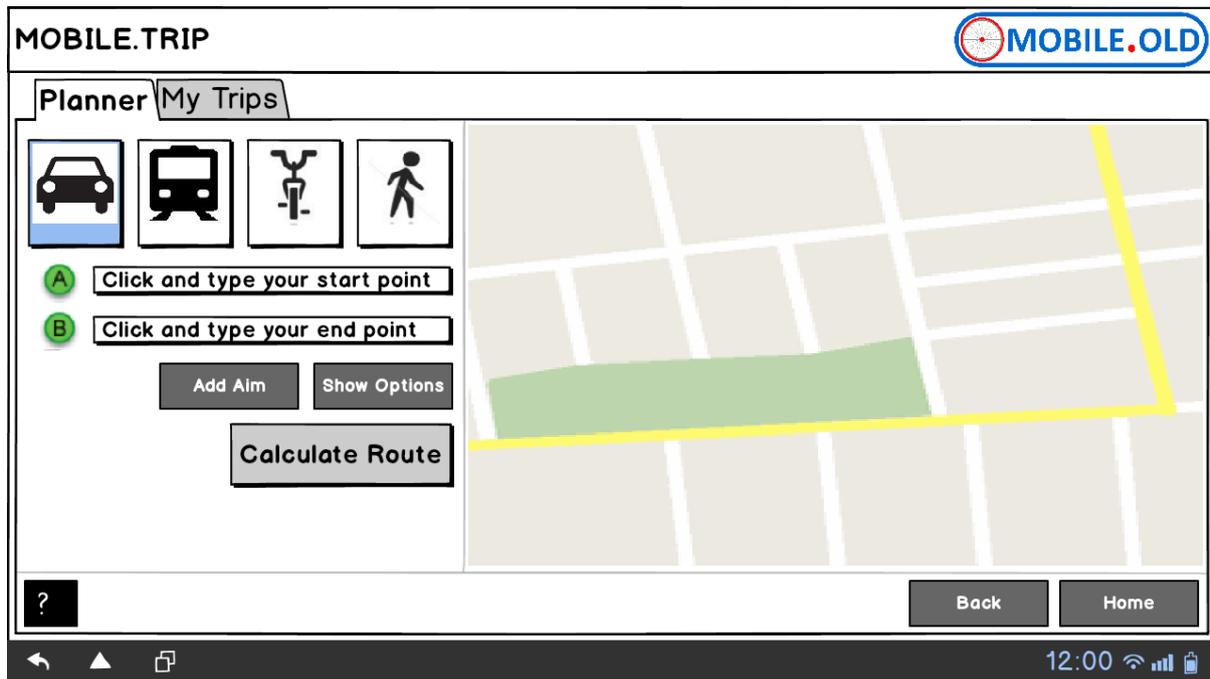


Figure 18: TJP Service - Home Screen on the TV

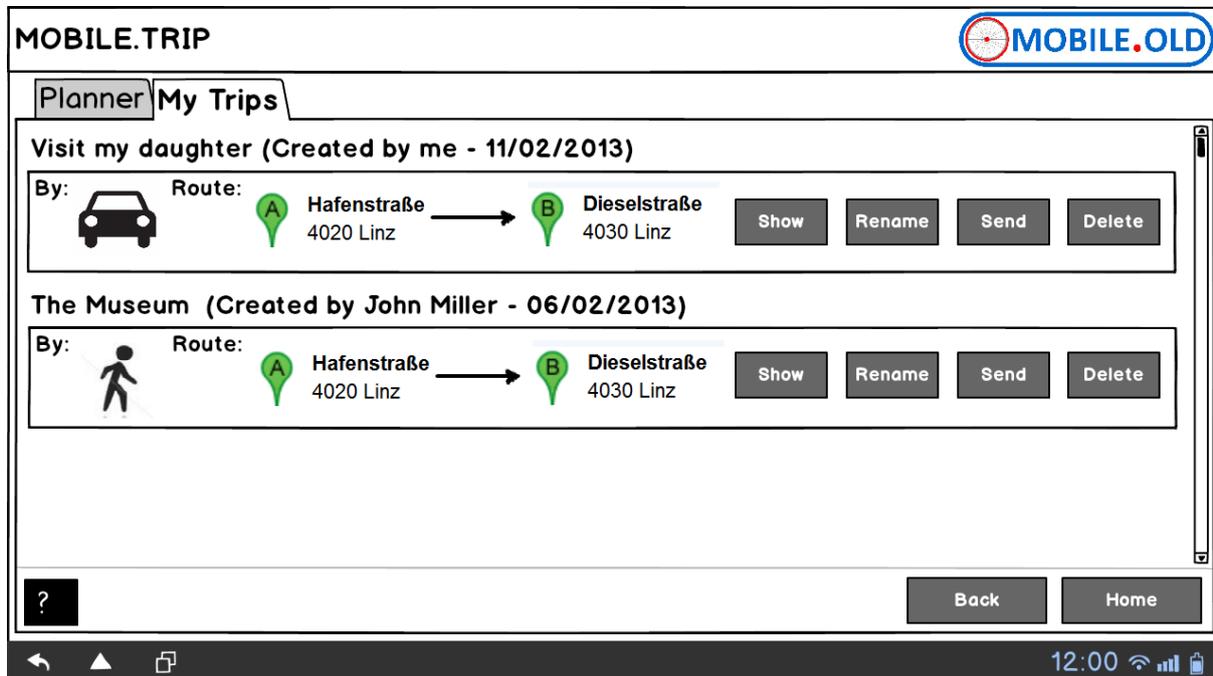


Figure 19: TJP Service - My Trips Screen on the TV

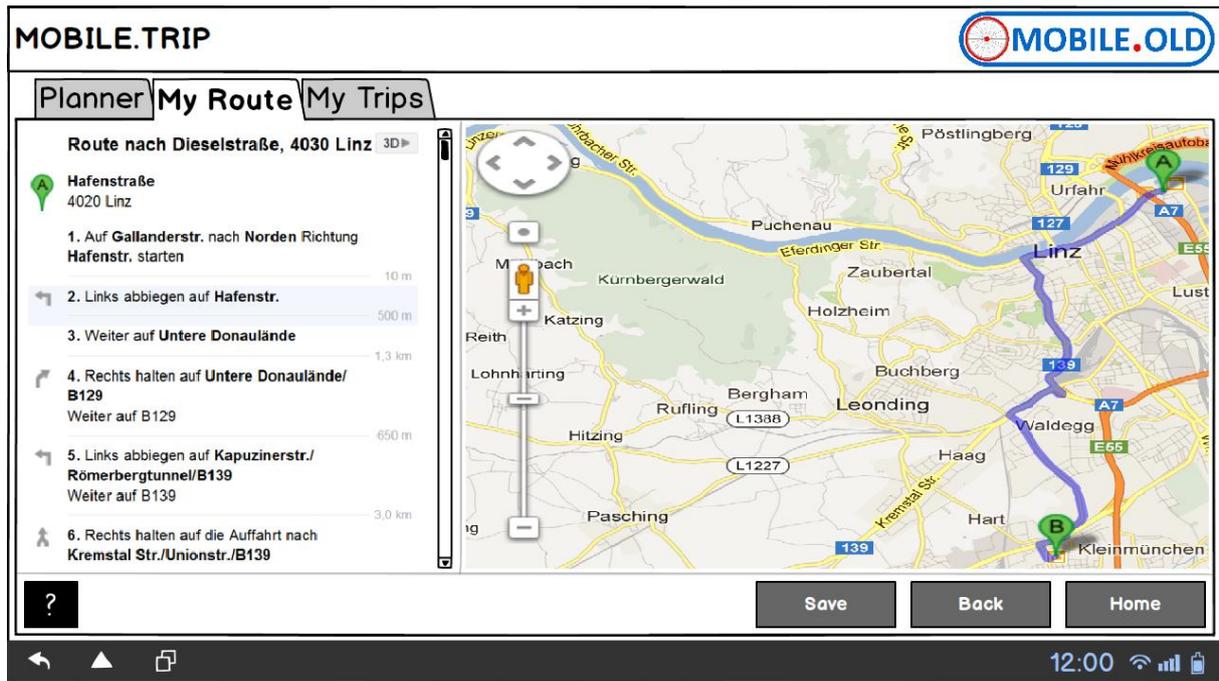


Figure 20: TJP Service - My Route Screen on the TV

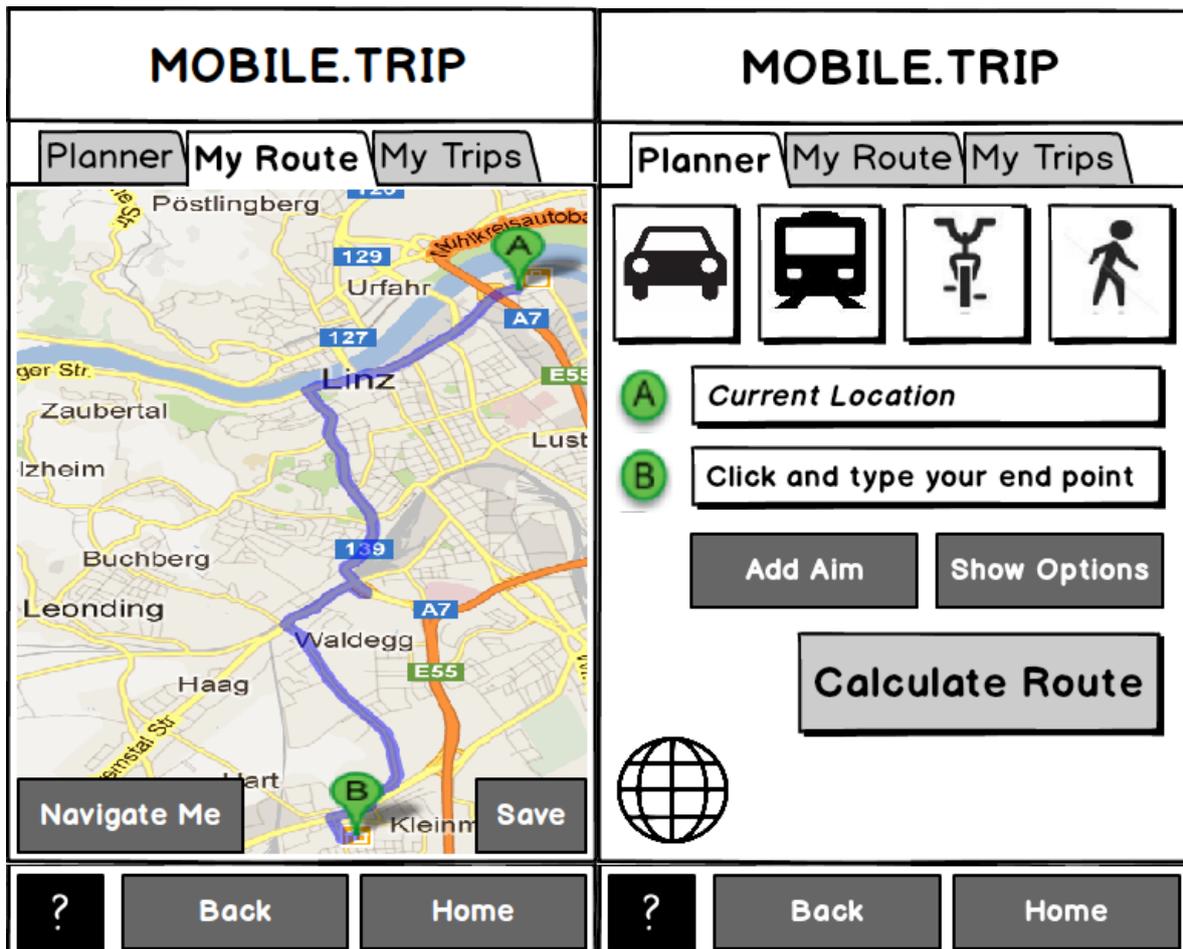


Figure 21: TJP Service - My Route & Home Screen on the Smart Phone

## 4.5 ORIENTATION AND MOBILITY TRAINING SERVICE

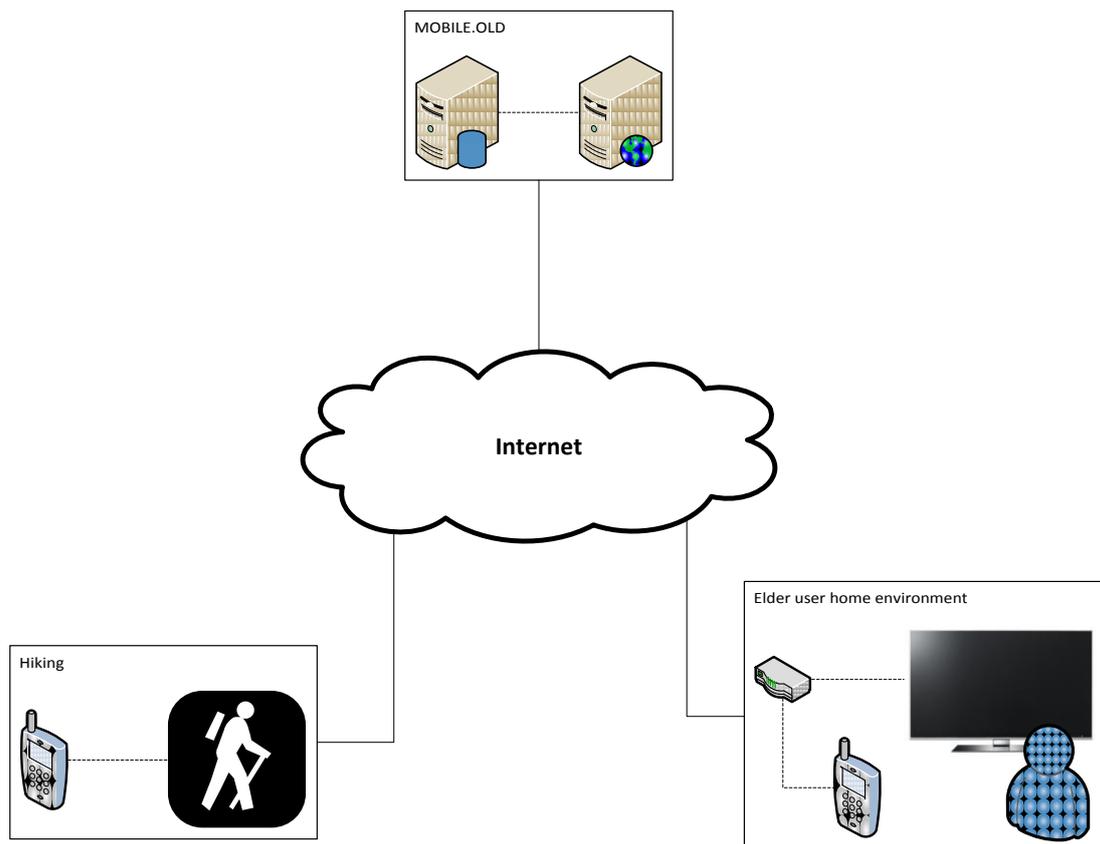
### 4.5.1 HIKING APPLICATION

#### 4.5.1.1 SERVICE OVERVIEW

Hiking application is part of MOBILE.OLD's Orientation and Mobility Training (OMT) service. It is an outdoor navigation, tracking and logging application ideal for hiking, biking, running, skiing, geocaching and other outdoor activities.

The elder person can record, import (via a e-mail or the TJP-Service) and follow tracks on his Smart Phone application and view his recorded track either directly on his mobile device or on his TV. Also, the user can share the recorded track or the collected information with his friends/family using E-mail. The elder users are able to see in real time the measures of time, distance, pace, speed, height, burned calories and elevation of their track.

The following figure shows a high level architecture diagram of this service:



**Figure 22: Hiking Application of the Orientation and Mobility Training service**

- Elder person uses his mobile device or TV (when in his home) and contacts the MOBILE.OLD service through the internet.
- The elder person uses his mobile device when he is out hiking, to record his track.
- After the hiking, the user can view his recorded track and information about it via his mobile device or TV.

#### **4.5.1.2 SERVICE CONFIGURATION**

The Hiking Smart Phone application allows the user to choose between standard GPS and, what kind of information to record as well as accessibility options which are part of the MOBILE.OLD Personalization Framework, such as contrast level, font size, etc.

#### **4.5.1.3 THE TV APPLICATION PROVIDES THE SAME CONFIGURATION PREFERENCES AS THE SMART PHONE APPLICATION. SERVICE OPERATION**

The main interface of the Smart Phone application is composed by an interactive map, a GPS strength indicator, a button to start tracking the trip of the user and a zoom control. The user is able to interact with the map with the standard Smart Phone gestures too (pinch, scroll, double tap).

When the user starts tracking, his trip is recorded and displayed on the map. After the user has recorded his track, it is automatically stored (under my tracks) and he can view the information about it or even share it using e-mail.

When the user returns home, he is able to visit the Hiking Application of the Orientation and Mobility Training service on his TV and view the track he just made (synchronized automatically) , along with the information about it and even share it with e-mail.

If the elderly user himself receives a track, via e-mail (by opening the attachment) on his Smart Phone or the TV, the track is automatically put to the my tracks section of the service.

If the user is following a track frequently, the senior is able to record his information every time he is on the track in order to compare the information, so that he is able to view his progress.

### 4.5.1.4 SERVICE MOCK-UPS

The following figure depicts a user interface mock-up of the Hiking Application.

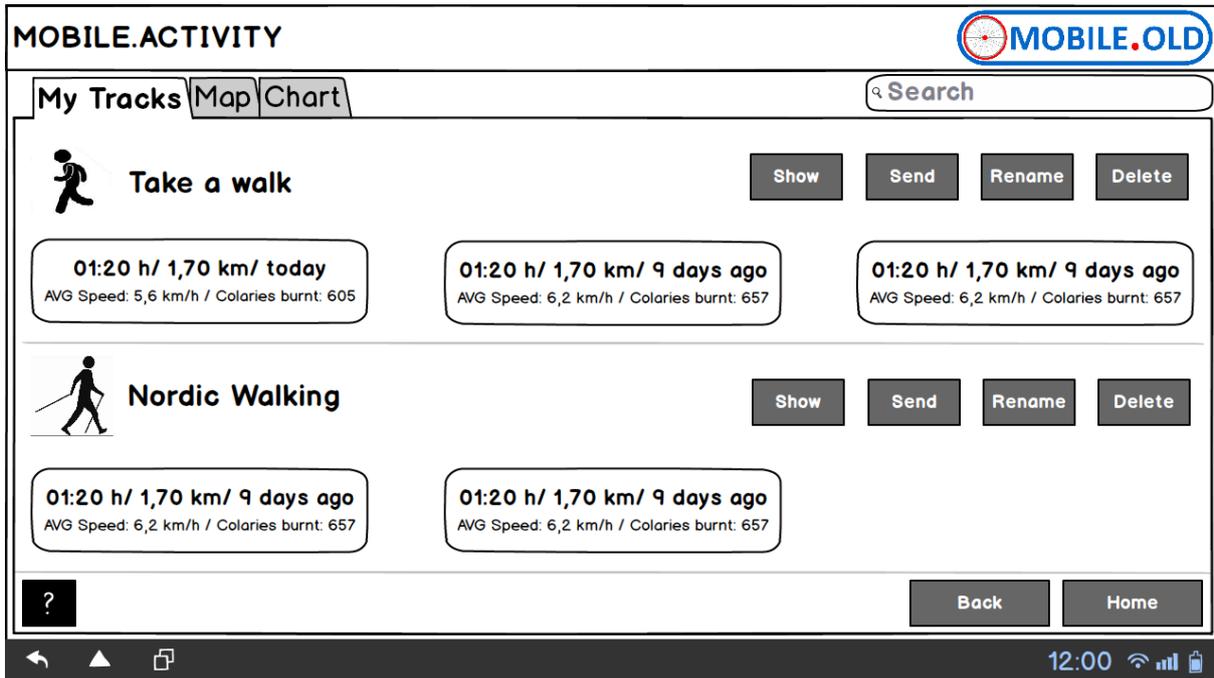


Figure 23: Hiking Application - Home Screen on the TV

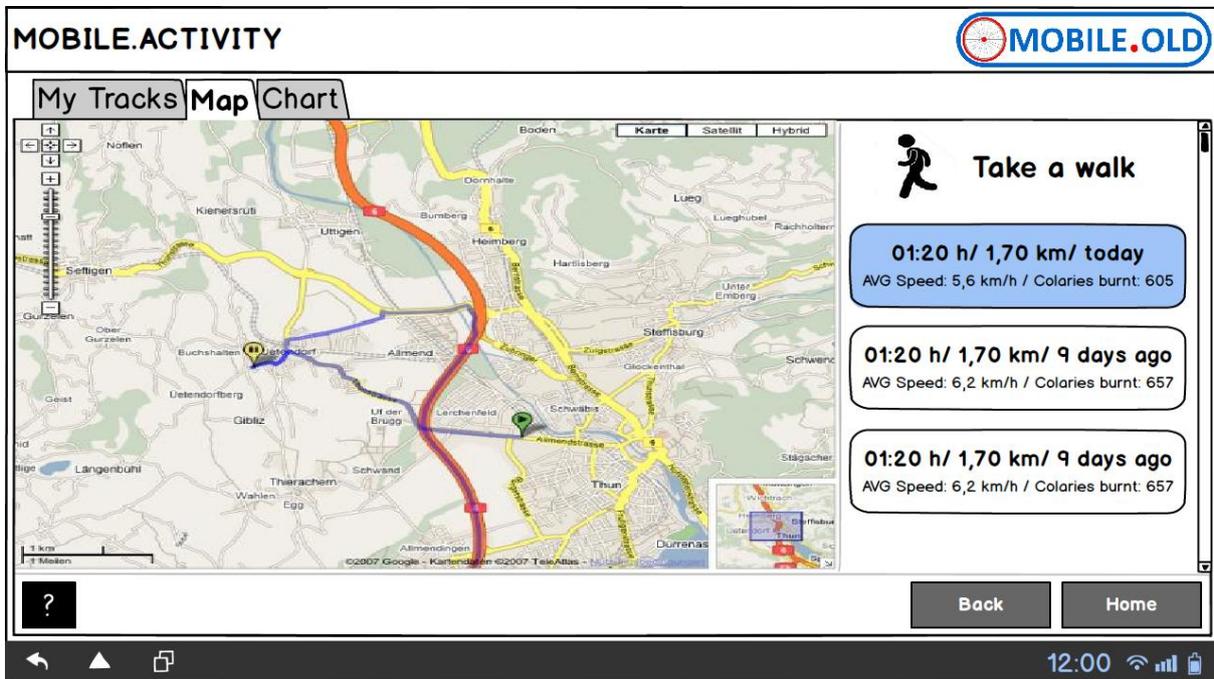


Figure 24: Hiking Application - Map Screen on the TV

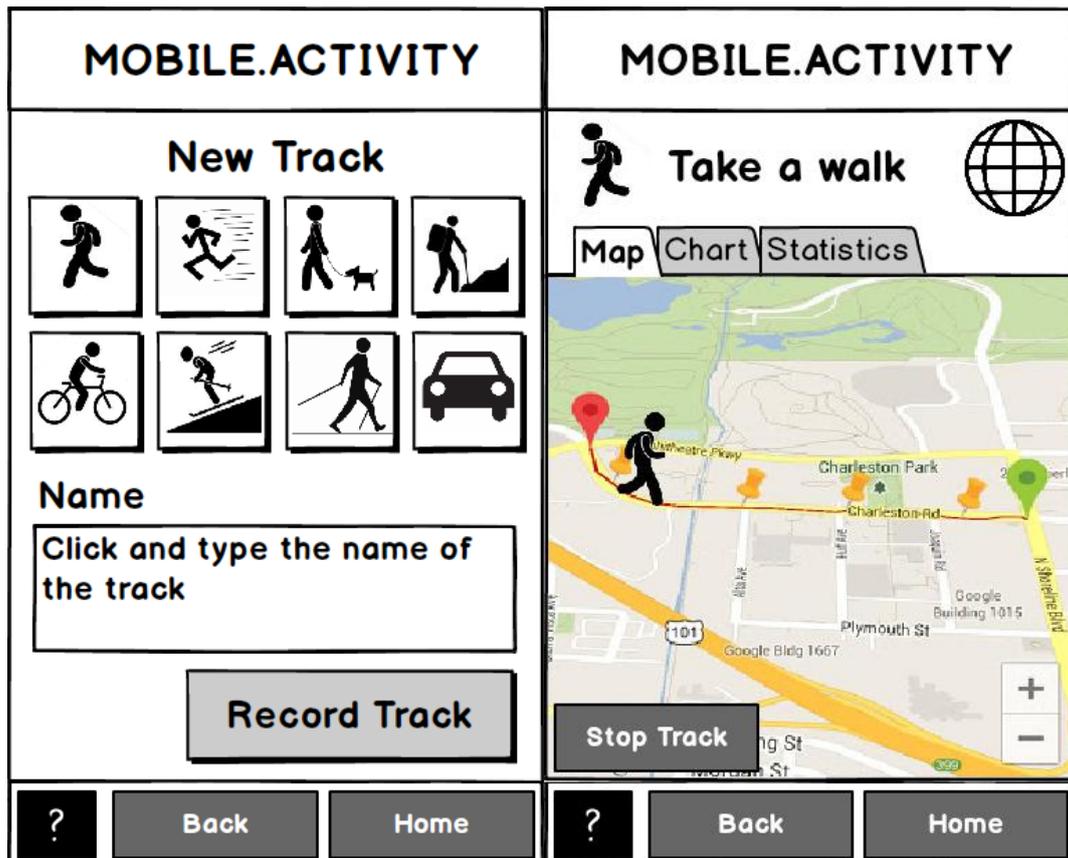


Figure 25: Hiking Application - New Track & Record Screen on the Smart Phone

## 4.5.2 ORIENTATION AND GEOTAG APPLICATION

### 4.5.2.1 SERVICE OVERVIEW

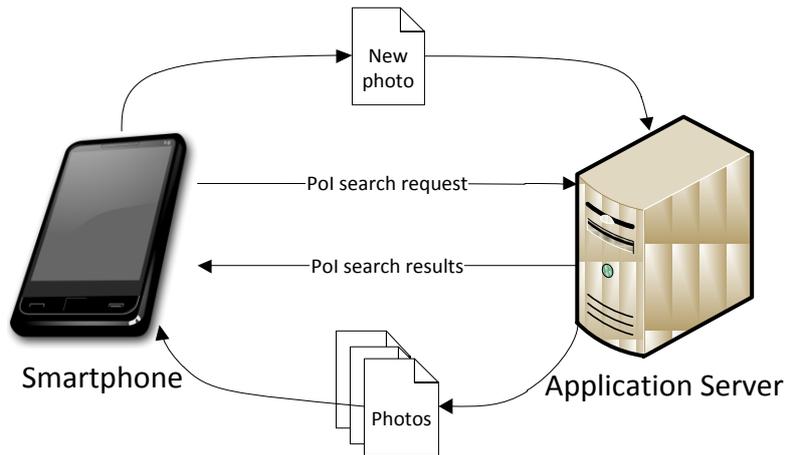
This service allows searching for certain kinds of locations, such as restaurants or train stations, as well as nearby points of interests (POI) in general. Upon selecting a destination, the application can provide a map-based route to the destination, including spoken instructions using TTS.

The app can also be used to take photos, which contain the current location coordinates as Exif metadata. In viewing mode, the photos' locations are designated as thumbnail-markers on a map, and can be viewed in full screen by touching the respective marker.

Maps, where used, are initially centred on the user's current location, with the relevant items in the surrounding area displayed as pins. Maps can be operated using standard swipe/pan gestures, as well as on-screen buttons for zoom, movement and reset (set map viewport and zoom level to default values).

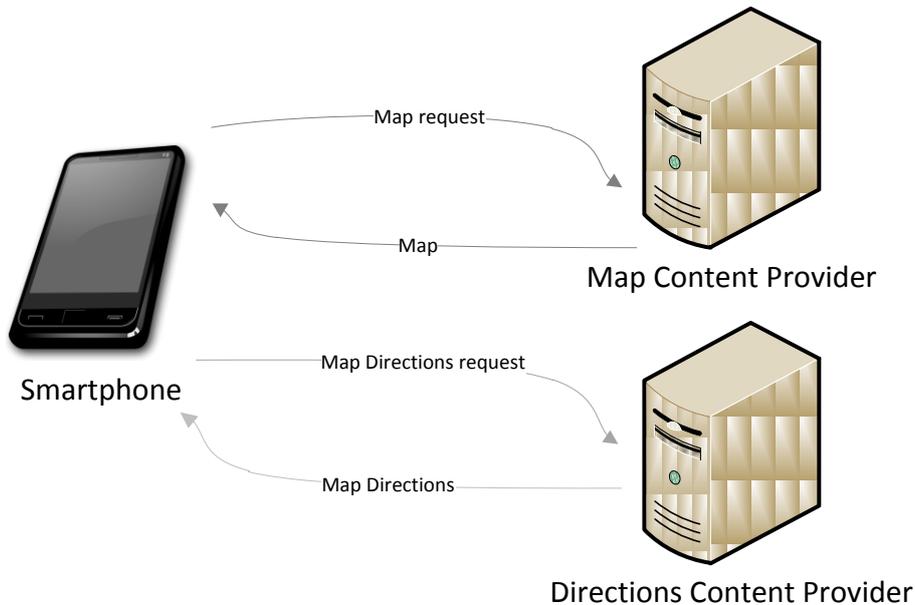
The only participating actor is the elderly user.

As visible in Figure 26, geotagged photos are persisted on an application server, and can be fetched from there if not cached on the client Smart Phone.



**Figure 26: Orientation and geotag application architecture**

When searching for POIs, a request is sent to the application server, which in turn queries other web services providing POI information. The processed result of those queries is sent back to the Smart Phone.



**Figure 27: Directions and Map Requests**

The application server provides the map data, shown on the device and is able to provide direction information between two points on the map. The data are extracted from service providers and relayed forward to the Smart Phone.

#### 4.5.2.2 SERVICE CONFIGURATION

Apart from global user settings, this application doesn't need specific configuration.

#### 4.5.2.3 SERVICE OPERATION

When starting the app, the user is presented with a menu allowing a choice between the following activities (Figure 28):

1. Search for nearby points of interest filtered by a category, which is selected after choosing this activity by either selecting one of the pre-defined category icons (such as restaurants, toilets, sights) or manually entering a custom category as text.
2. Search for nearby points of interest of any category
3. Show recently selected locations
4. Show previously defined locations
5. Take a geotagged photo
6. View geotagged photos on a map

### Map

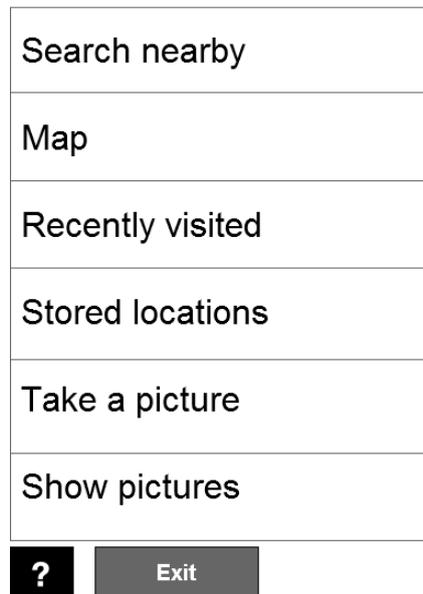
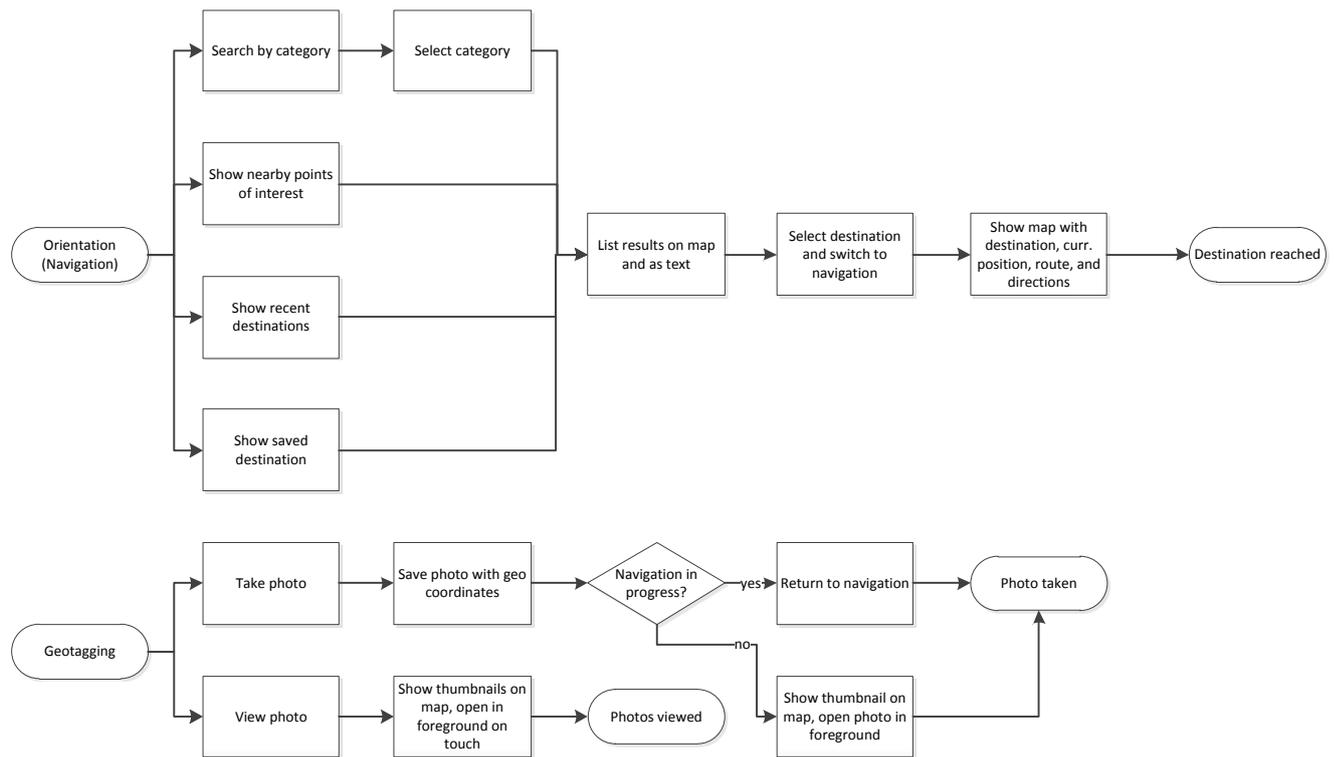


Figure 28: Orientation Service: Main menu

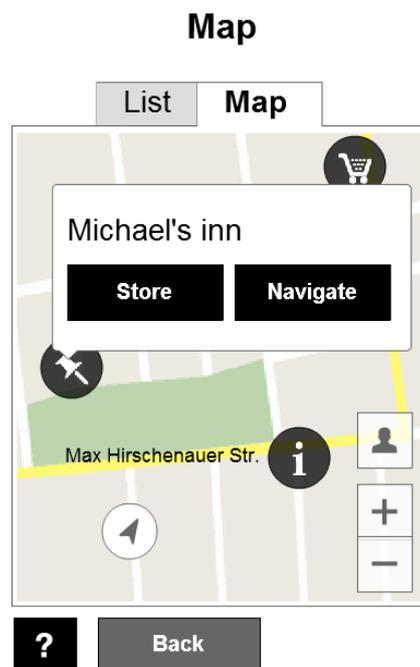


**Figure 29: Operation of the orientation and geotagging application**

Selecting any of the activities 1 through 4 results in a map with pins (see Figure 31) identifying the points of interest, as well as a list of them (see Figure 30). Selecting a point of interest highlights it on the map. If available, the pins show a thumbnail of a photo of the location. If available, the users’ own (geotagged) photos are preferred over photos from the Internet.



**Figure 30: Orientation Service: List View**



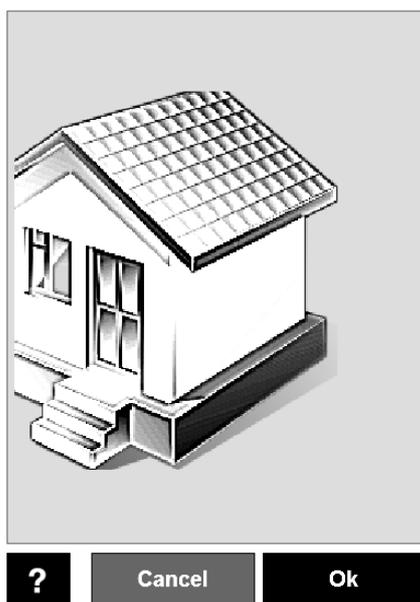
**Figure 31: Orientation Service: Map View**

Once a destination has been selected by touching either a list item or a pin, navigation can be started. In this case, a map showing the user's current location, the destination, and a route to it covers most of the screen. The route map is complimented by a list of walking directions, which can optionally be spoken aloud using TTS.

When option 5 is selected, the user is provided with a minimalist interface to take a photo (Figure 32). Upon taking it, the user is redirected to a map with the recently taken photo is displayed as a highlighted thumbnail pin.

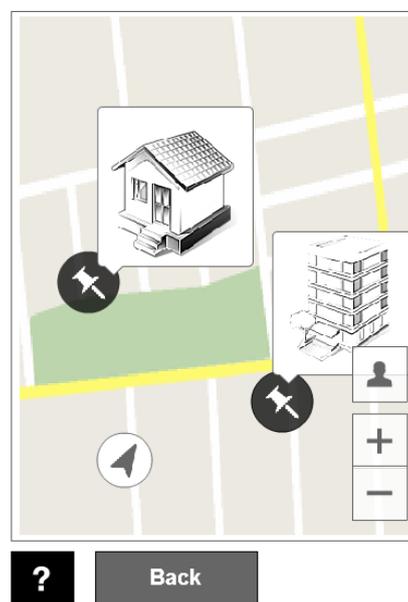
When viewing geotagged photos (option 6), all nearby photos are showed on a full screen map as thumbnail pins (Figure 33). When touched, the full-size image is displayed in the foreground.

**Take picture**



**Figure 32: Orientation Service: Camera Viewfinder**

**View pictures**



**Figure 33: Orientation Service: Showing taken photos on map**

### 4.5.3 INDOOR / OUTDOOR ORIENTATION TRAINING APPLICATION

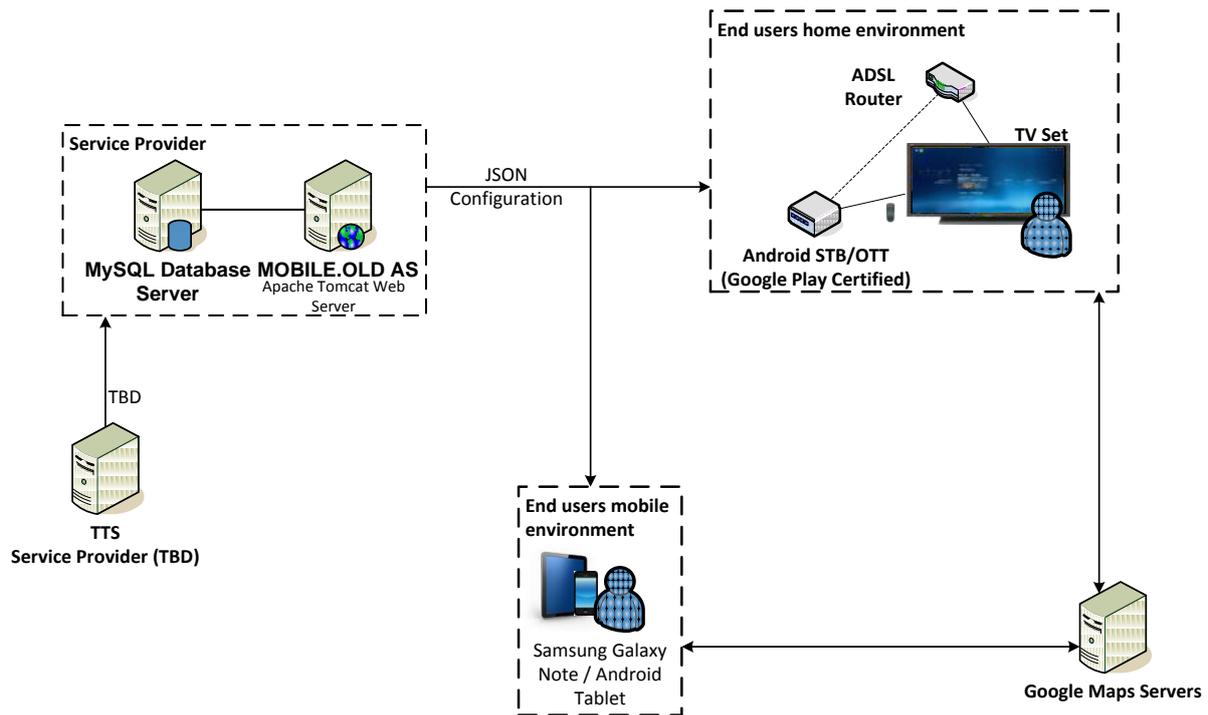
#### 4.5.3.1 SERVICE OVERVIEW

The service provides indoor and outdoor photos of various facilities (such as museums, landmarks, etc.) around the world and allows the user to virtually tour them from his TV. The service uses Google's Street View technology to provide indoor and outdoor views of selected facilities, allowing the user to navigate them.

The service is accessible via TV application and displays a list of the available points of interest categorized based on their type (e.g. museums, shopping malls, physical landmarks, etc.). The user is able to select a POI from the list and the application will display its indoor or outdoor view depending on its type.

The service uses a third party service to retrieve the indoor and outdoor views and thus, it requires an active internet connection to function.

The following diagram depicts a high level architecture of the service.



**Figure 34: Indoor / Outdoor Orientation Training application high level architecture**

The overall features of the application are the following:

- Indoor and outdoor views for various buildings/facilities such as airports and shopping malls.
- Indoor and outdoor virtual touring of the selected point of interest.

#### 4.5.3.2 SERVICE CONFIGURATION

The application allows the user to select the street view collection that he wants to be displayed by providing display filters.

#### 4.5.3.3 SERVICE OPERATION

The main interface of the application is composed by a list of street view collections the user can virtually tour to. From the main interface, the user can select a collection and be presented with the available Views for the selected category. Each View entry has an indication depicting whether it provides an indoor or an outdoor view of the respective facility. The user can select a View and be presented with an impression of the facility, navigable via the TV remote control.

### 4.5.3.4 SERVICE MOCK-UPS

The following figures depict user interface mock-ups of the Indoor / Outdoor Orientation Training service.

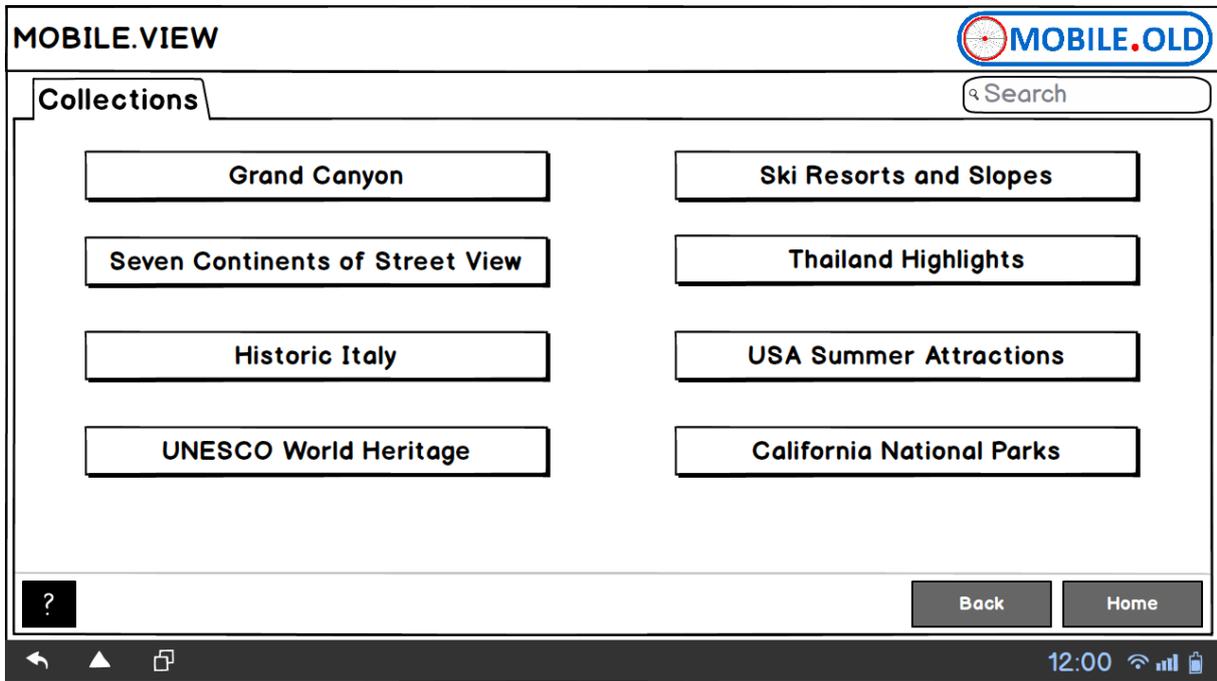


Figure 35: Indoor / Outdoor Orientation Application - Main Interface on the TV

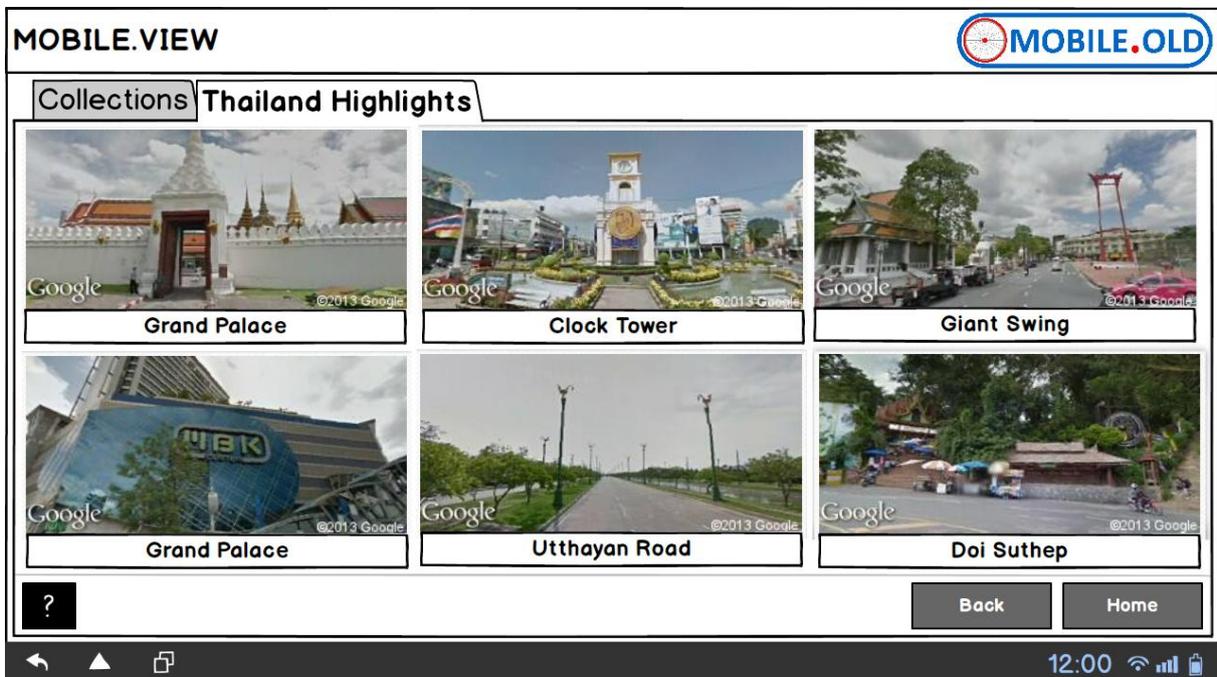


Figure 36: Indoor / Outdoor Orientation - Collection Screen on the TV



Figure 37: –Google Street View Application on the TV

#### 4.5.4 FIRST AID APPLICATION

##### 4.5.4.1 SERVICE OVERVIEW

The first aid application primarily provides guides to handling certain health emergencies. Also, the phone numbers of the national emergency services (police, ambulance, fire brigade) in the country the user currently is in are displayed. Due to legal restrictions, a quick dial option is not available for emergency numbers.

The involved actors are:

- **The elderly user** as the person consuming the information provided by the application.
- **The service provider**, who is responsible for maintaining the emergency guides by creating new ones or adapting existing ones, as well as adding emergency numbers.

The first aid guides don't take priority over the professional opinion of a trained medical practitioner. Therefore, a disclaimer should be visible upon the first launch of the application, declining all liability if a person was harmed related to the first aid application.

The available guides should cover the most important emergency situations, but their number should be very limited, so the relevant guide can be found quickly when actual emergency situations occur.

Compared to existing first aid application, the focus and big advantage of this service is the intuitive and easy to use user interface.

As shown in Figure 38, the Smart Phone queries the application server for updated first aid scenarios. The results are fetched, processed and shown on the Smart Phone.

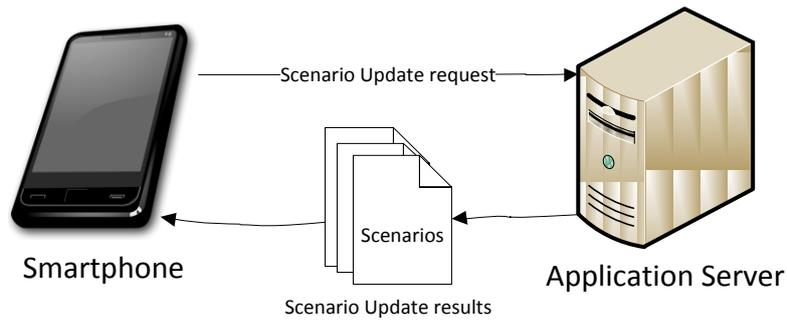


Figure 38: First aid application architecture

4.5.4.2 SERVICE CONFIGURATION

The elderly user may pin guides they consider important to the application’s main menu, so they can be accessed quickly at the touch of a button. The guides which are available through a shortcut in the main menu by default are determined by tracking the general popularity of the guide, considering **weights** that were defined manually by the service provider. A first aid scenario is stored as a sequence of **textual** descriptions with optional **pictures**.

4.5.4.3 SERVICE OPERATION

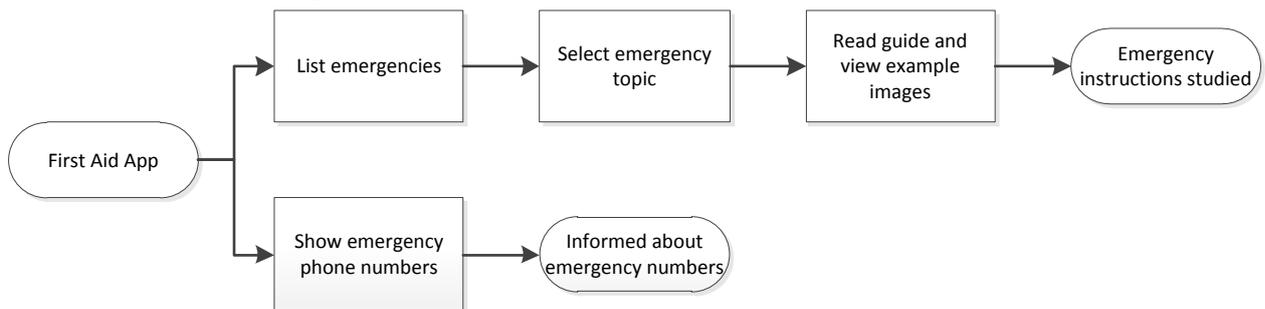


Figure 39: Operation of the first aid application

After launching the orientation service, a main menu (Figure 40) is presented. The user chooses to either view emergency phone numbers (Figure 42), or an emergency guide (Figure 41). In case of the latter, the user first selects the relevant category, and then the applicable guide.

### First aid

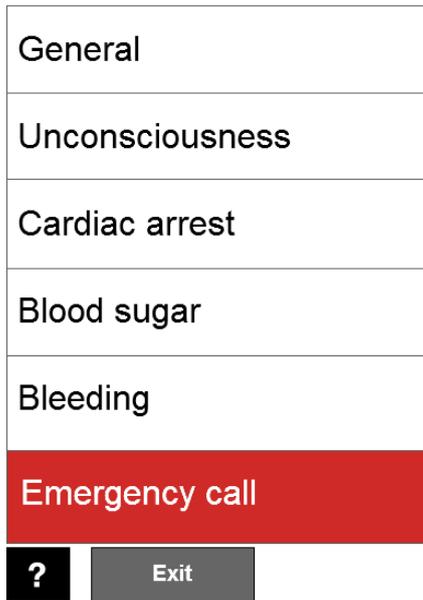


Figure 40: First Aid: Main menu

### General



Figure 41: First Aid: Emergency Guide

### Emergency call



Figure 42: First Aid: Emergency numbers

### Emergency call

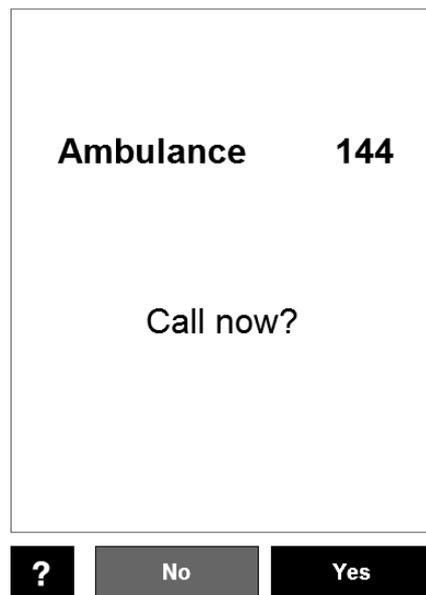


Figure 43: First Aid: Call emergency number

## 4.5.5 CHECKLIST APPLICATION

### 4.5.5.1 SERVICE OVERVIEW

The checklist application allows creating, editing and reviewing checklists and checklist templates, using predefined or custom checklist items. Custom items can be either text, a picture (it is possible to take photos from within the app), or audio (it is possible to record voice from within the app), whereas it's possible to specify alternative text when using a picture or audio as medium.

Among the checklists to be included by default are:

- Vacation essentials
- Shopping essentials (generic example for the elderly to build on)

Checklists can also be used to keep track of medication. For example, if the user loses some important medication in a foreign country, they can show a picture taken with the checklist app to a pharmacist to receive an equivalent replacement.

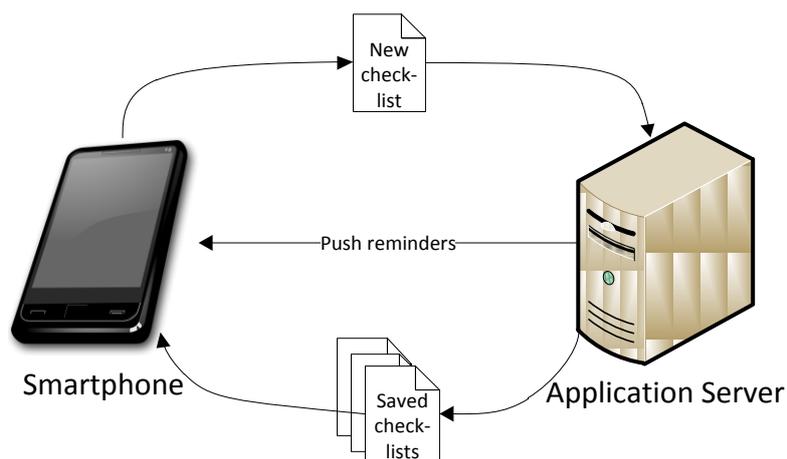
It is possible to read the checklist out loud using TTS.

The checklist supports time- and location-based reminders. When choosing the latter, the user associates a location with the checklist. The next time the user visits that location, they are reminded of their checklist.

The involved actors are:

- **The elderly user** as the person who creates and uses checklists.
- **The service provider** as the creator and maintainer of the default template checklists.

One of the main advantages of this app compared to classic pen and paper checklists is the intuitive, easy to use user interface.



**Figure 44: Architecture of the checklist application**

As visible in Figure 44, checklists are persisted on the application server. Reminders can be sent from the application server as push notifications.

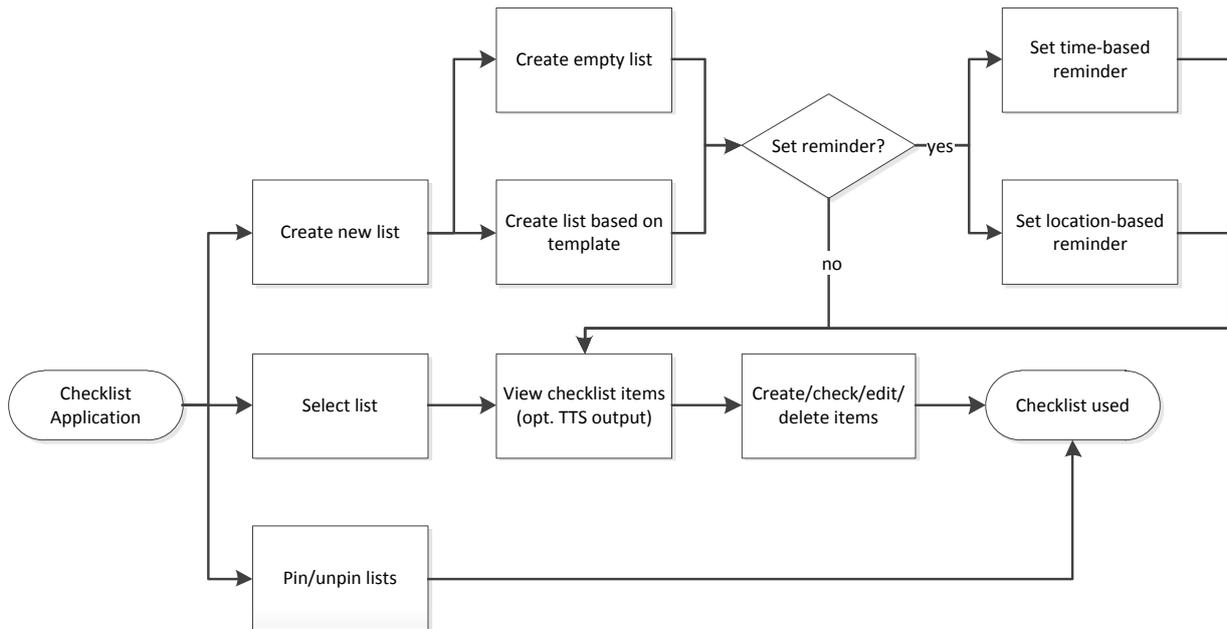
**4.5.5.2 SERVICE CONFIGURATION**

The service provider can define checklists which are available in the application by default.

The user can influence the order of displayed checklists by pinning them to a certain position.

**4.5.5.3 SERVICE OPERATION**

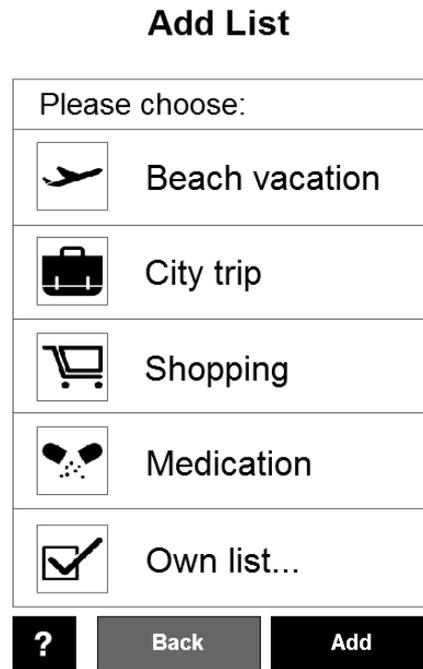
When starting the application, the user sees a list of all checklists (Figure 46). They are ordered so that the most recently used check list is on top. The user can “recycle” a list. This function enables the reuse of a checklist by simply unchecking all items on it.



**Figure 45: Operation of the checklist application**



**Figure 46: Checklist: Main view**



**Figure 47: Checklist: Choosing a new list**

When choosing to create a new list (Figure 47), the user can either select an existing list as a template, or create an empty list. In either case, the user chooses an icon from a pre-defined set of easily discernible icons to identify the list (a textual description may be added optionally). It is also possible to optionally set a reminder (Figure 48). After creating the list, the user is redirected to the new list itself. The list is initially empty (Figure 49). To add items to the list, the user can choose between predefined ones (according to the selected template) or “static” ones that can be of the type text, photo or audio (Figure 50). After successfully creating an item, the user returns to the associated checklist, which now contains the new item. Checking an item will place it at the bottom of the screen (Figure 51).

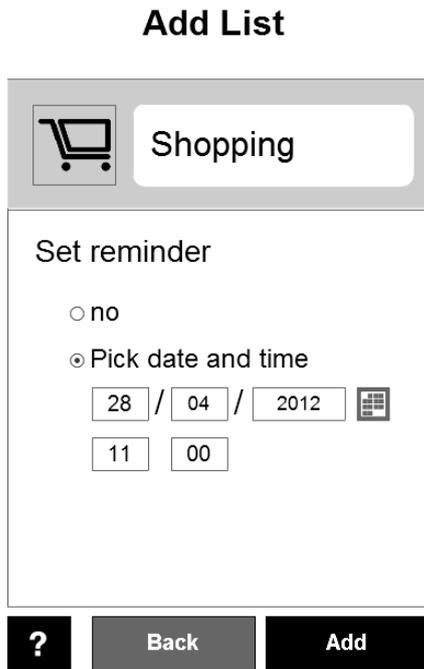


Figure 48: Checklist: Adding a list with reminder

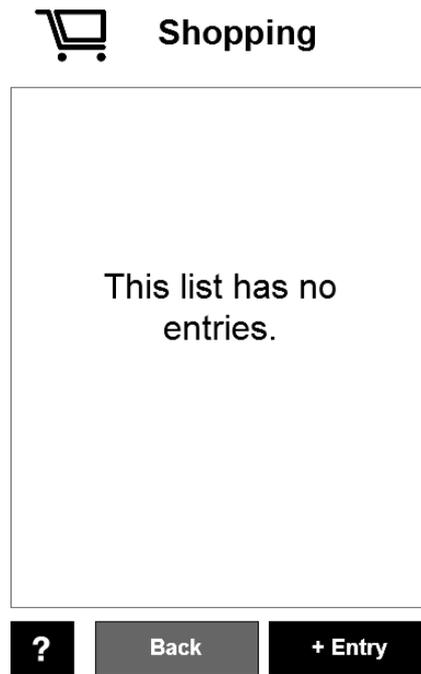


Figure 49: Checklist: The new and empty list

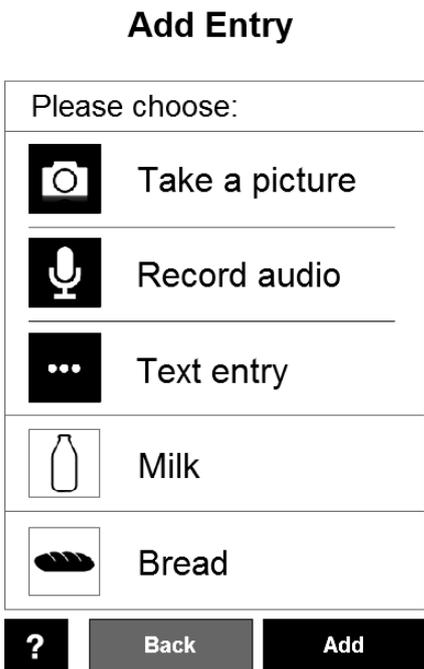


Figure 50: Checklist: Adding an entry



Figure 51: Checklist: List with (checked) items

## 4.5.6 ORIENTATION AND MOBILITY GAMES

### 4.5.6.1 SERVICE OVERVIEW

The main part of the orientation and mobility games service is a Scavenger Hunt game. This is a game where the player has to go to a certain place in order to find clues to the location of the subsequent clues which ultimately lead to a goal.

Other games may be created as well, such as a memory game or a first aid quiz.

The involved actors are:

- **The elderly user** as the person who plays games.
- **The service provider** as the creator and maintainer of the game data.

The scavenger hunt is a playful way to motivate the user to visit a sequence of destination, which are either defined by the service provider or the user. A sequence of destinations must be visited in order to complete the scavenger hunt. A Score system in measures the time it took the user to reach a place and the distance travelled. Based on this data, a score is generated in order to motivate the user.

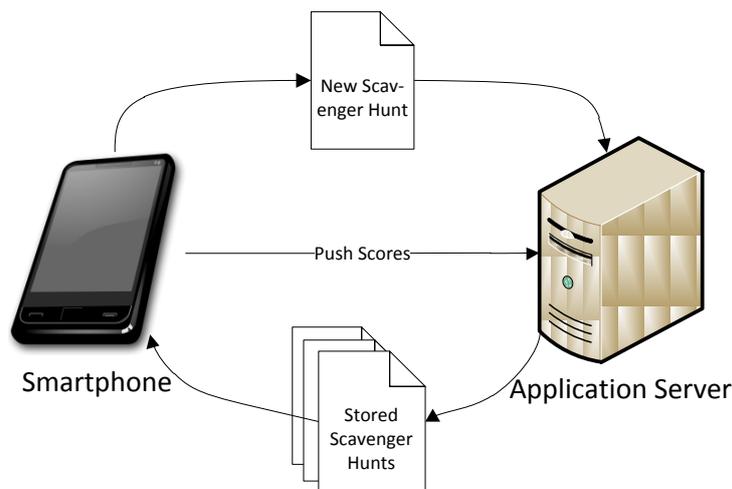


Figure 52: Scavenger Hunt game architecture

#### 4.5.6.2 SERVICE CONFIGURATION

Scavenger hunts can be pre-defined by the service provider. Users can also create their own scavenger hunts by selecting a sequence of destinations on a map, or by adding a destination to the hunt while they are there, so they can re-visit it later. A scavenger hunt configuration consists of a sequence of locations (as **coordinates**) and **textual** descriptions. The **scores** are pushed and stored on the server.

#### 4.5.6.3 SERVICE OPERATION

General operation of the game works as illustrated in Figure 53. Note that a separate game launcher application is omitted to avoid an additional navigation step.

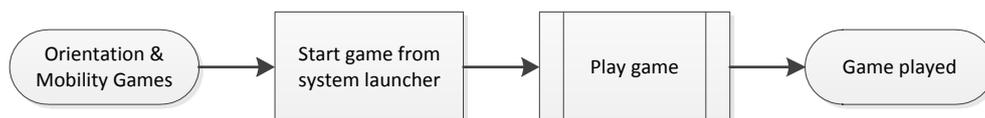


Figure 53: General procedure for playing an orientation and mobility game

Figure 54 demonstrates the operation of the scavenger hunt. When the application is started, the user sees a main menu (Figure 55). There, the user can create a new game and add custom places (Figure 56). Generally, performing a scavenger hunt means that the user is shown a destination (Figure 57) and as soon as it is reached (Figure 58), he or she has to answer questions (Figure 59). After that, the next destination is show on the game screen. This is going on until the previous one was the final destination. If the user has trouble reaching the destination, they can switch to the navigation app to receive a route and instructions. The main menu also allows the player to review finished games and including the answered questions.

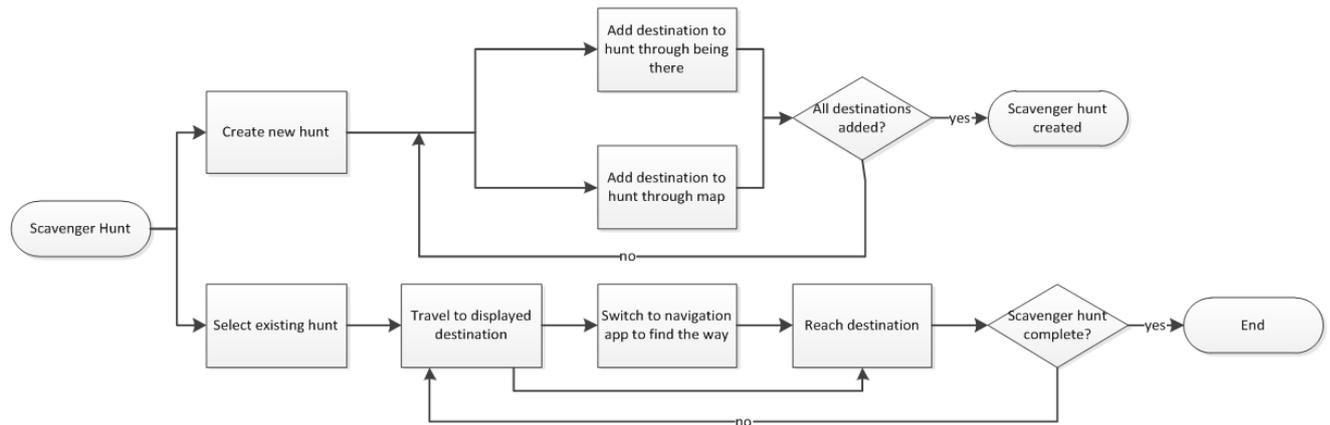


Figure 54: Operation of the scavenger hunt game

**Mobile Quiz**

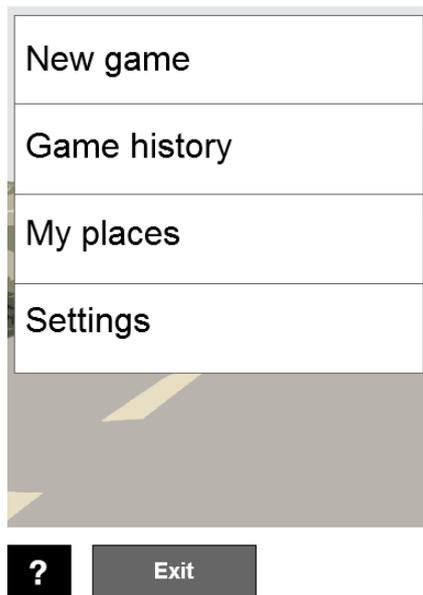


Figure 55: Mobile Games: Main Menu

**New place**

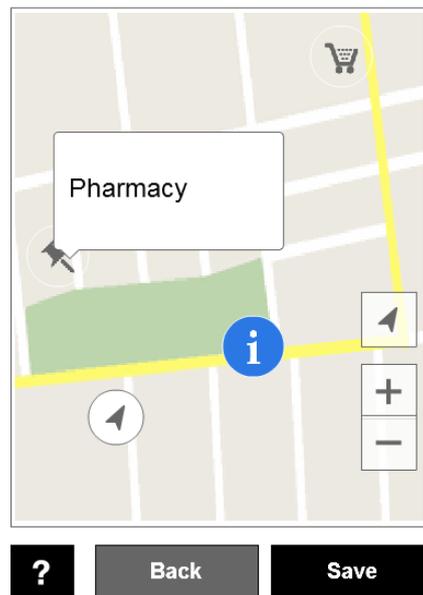


Figure 56: Mobile Games: Adding a destination

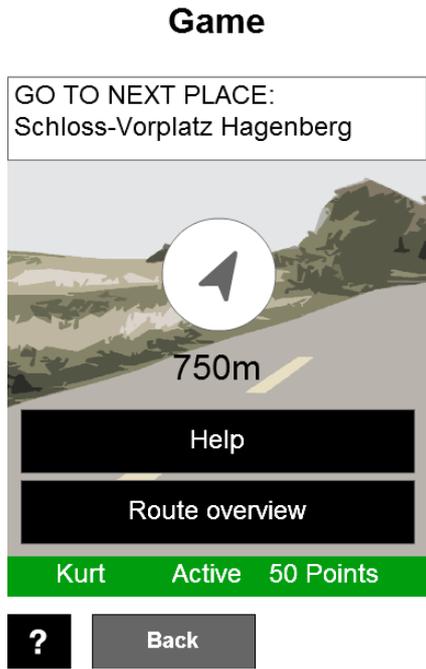


Figure 57: Mobile Games: Game screen

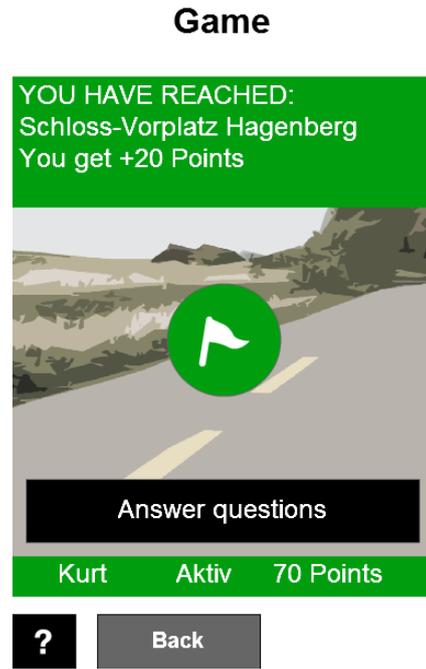


Figure 58: Mobile Games: Destination reached

### Question

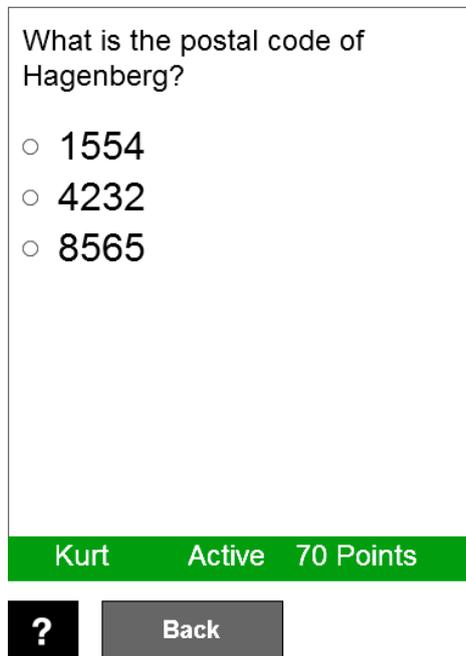


Figure 59: Mobil Games: Question

## 5. SUPPORTED END-USER TERMINALS

### 5.1 TV PLATFORMS

The MOBILE.OLD services will target multiple platforms such as TVs and mobile devices. Specifically, the MOBILE.OLD services will be implemented on top of the Android platform which features a wide range of devices.

The following table depicts indicative Android over the top boxes that will be able to be supported by the services initially.

Model	Android version	CPU	RAM	Connectivity	Storage	TV out
Hisense Pulse Google TV	3.2	1.2GHz	1GB	Ethernet, WiFi, USB	4GB	HDMI
Dx.com OEM	2.3	1GHz	512MB	Ethernet, WiFi	4GB	HDMI, VGA
FVA Android TV Box	4.0	1GHz	512MB	Ethernet, Wifi	2-4GB	HDMI
FVD-11 Androstick	4.0	1GHz	512MB	WiFi	4GB	HDMI
TomTop Android STB	4.0	1GHz	512MB	Ethernet, WiFi	2GB	HDMI
Vizio Co-Star Google TV	3.2	1.5GHz, Dual core	1GB	Ethernet, WiFi, USB, Bluetooth	4GB	HDMI
Sony Internet Player with Google TV	3.2	1.2GHz, Dual core	1GB	Ethernet, WiFi, USB	8GB	HDMI
Minix Neo X5	4.1.1	1.6GHz, Dual core	1GB	Ethernet, WiFi, USB	16GB	HDMI

**Table 1: Android Over The Top (OTT) devices**

The selected Android Over The Top device that will be used to develop and validate the MOBILE.OLD services is the Minix Neo X5 Android OTT, which features the latest Android version, can connect to the Internet and has plenty of processing power (1.6GHz), RAM (1GB) and available storage (16GB).

## 5.2 SMART PHONE PLATFORMS

The mobile endpoints of the MOBILE.OLD services will target the Android platform and will be design for optimal usage on the Samsung Galaxy Note Smart Phone which features a bigger than normal screen display and received positive comments from the user during the D2.1 activities.

Although the mobile applications of the services will be optimized for the Galaxy Note Smart Phone, they will be able to be used by any device which features Android 2.3 or earlier, has at least 1GHz CPU and 512MB of RAM.

## 5.3 STBs

Set-Top-Boxes feature similar functionality with over the top devices but retrieve the content through a different channel than the Internet. In this context, any Android STB will be able to be supported by the MOBILE.OLD services, with minor to none modifications. The MOBILE.OLD services will support the SAFEVIEW IPO-85X Android STB which features a 2500MIPS processor, full HD H.264 decoding, WiFi, Android 4 and a WebKit based web browser.

## 5.4 TABLETS

As described, the mobile endpoints of the MOBILE.OLD services will target the Android platform, thus they will be able to support any Android based tablet device.

The following table depicts indicative Android tablet devices that will be able to be supported by the MOBILE.OLD services.

Model	Android version	Screen size	CPU	RAM	Connectivity	Storage	GPS
Google Nexus 7	4.1	7.0 in	1.3 GHz Quad	1GB	WiFi Bluetooth USB 2.0	16-32GB	Yes
Google Nexus 10	4.2	10.5 in	1.7GHz, Dual core	2GB	WiFi Bluetooth Micro USB	16-32GB	Yes
Samsung Galaxy Tab	2.2	7.0 in	1 GHz	512MB	WiFi Bluetooth 3G/4G available	16GB internal, microSD slot	Yes
Samsung Galaxy Tab 2	4.0	7.0 in	1 GHz Dual core	1GB	Wifi Bluetooth 3G/4G	8-16GB internal, microSD slot	Yes

					available		
Lenovo Ideapad	4.0	7.0 in	1GHz	512MB	WiFi Bluetooth	8GB internal, microSD	Yes
Acer Iconia Tab A110	4.1	7.0 in	1GHz Quad	1GB	WiFi Bluetooth	8GB internal, microSDHC	Yes

**Table 2: Android Tablet Devices**

Since the services will be developed on top of the Android platform, any Android tablet device will be supported. However, the services will be optimized to run in either the Google Nexus 7 or the Google Nexus 10 tablets. The final decision will be made after the first services trials.

## 6. SW TECHNOLOGIES SELECTION

### 6.1 HUMAN MACHINE INTERFACES TECHNOLOGIES

The interaction with the Smart Phone can be divided in two main components depending on the specific requirements of the MOBILE.OLD services:

- the elder needs to interact with the MOBILE.OLD application using the mobile device or TV (send/receive information, receive notifications, etc.), see for example the personalized transportation news headlines service or the orientation and mobility training service;
- the MOBILE.OLD application monitors the state of the elder using the Smart Phone sensors, see for example the geofencing service or the orientation and geotag application.

In the first case, for the mobile device and TV, the touchscreen and the remote control will be the main interface. The graphical user interface, which has to be simple and easy to handle, will be developed using native Android APIs and technologies and the TV GUI will be designed using the 10-foot user interface principles.

For the second situation the Android API will be used to interact with the Smart Phone sensors. The Android API provides methods to acquire information from the following types of sensors: accelerometer, ambient temperature, gravity, gyroscope, light, linear acceleration, magnetic field, orientation, pressure, proximity, relative humidity, rotation vector and temperature. For Smart Phone location detection the Google Location Services API will be used.

The state-of-the-art analysis and the HOMEdotOLD experience have shown that several TTS engines are available in the market either as free or commercial services. The selection of the TTS engine is not possible at this stage since the different available solutions require further analysis and experimentation through prototyping. Such an approach is mandatory in order to be able to select the one that provides the End User with high Quality of Experience. In this context, the activities towards the selection of the TTS engine will be performed during the detailed design phase.

During the state-of-the-art analysis, a set of popular gesture recognition platforms for TV have been identified including Wii Remote and Microsoft Kinect. Nevertheless, these solutions provide conditional to no compatibility with TV-based platforms and their integration requires significant effort, extra hardware and design workarounds to accommodate the interfaces with those devices. In this context, such gesture recognition platforms are not currently considered for integration in MOBILE.OLD. Nevertheless, this technology will be monitored throughout the whole lifecycle of the project and if their compatibility with the TV platforms is more mature, they might be reconsidered for integration.

The main input interface of the End Users with the TV set will be the remote control since this is the main interface of the TV set and the services will be designed by primarily taking this into consideration. Nevertheless, since the end users prefer keyboard and mouse as the main alternative, the MOBILE.OLD services will ensure full compatibility with these input devices.

## 6.2 COMMUNICATION TECHNOLOGIES

The MOBILE.OLD platform will realize its external interfaces through AJAX and web services.

AJAX (Asynchronous JavaScript and XML) is a set of technologies and web development techniques used on the client side to create asynchronous web applications. AJAX permits web applications to send and receive data from a web server asynchronously without interfering with the display and behaviour of the existing page. To achieve the desired functionality, AJAX incorporates the following technologies:

- HTML and CSS for presentation.
- The Document Object Model for dynamic display and interaction with the data.
- XML for the interchange of data and XSLT for its manipulation.
- The XMLHttpRequest object for asynchronous communication.
- JavaScript to bring the aforementioned technologies together.
- Describe the communication technologies that will be used for the platform's external interfaces (e.g. SOAP, AJAX)

Web services are a method of communication between two devices over the World Wide Web. A web service is a software function provided at a network address over the web or the cloud and is designed to support interoperable machine-to-machine interaction over a network. Web services feature an interface described in a machine process-able format (a Web Services Description Language).

## 6.3 DATABASE TECHNOLOGY

MOBILE.OLD will use a MySQL database to store the user preferences and the service's configuration.

MySQL is an open source relational database management system that runs as server software providing multi-user access to a number of databases. A relational database stores data in separate tables rather than one big storeroom and the database structures are organized into physical files optimized for speed. The logical model, with objects such as databases, tables, views, etc. offers a flexible programming environment, allowing the setup of rules governing the relationships between the different data fields and the database enforces them.

The MySQL database software is a client/server system that consists of a multi-threaded SQL server that supports different client programs and libraries, administrative tools and a wide range of application programming interfaces.

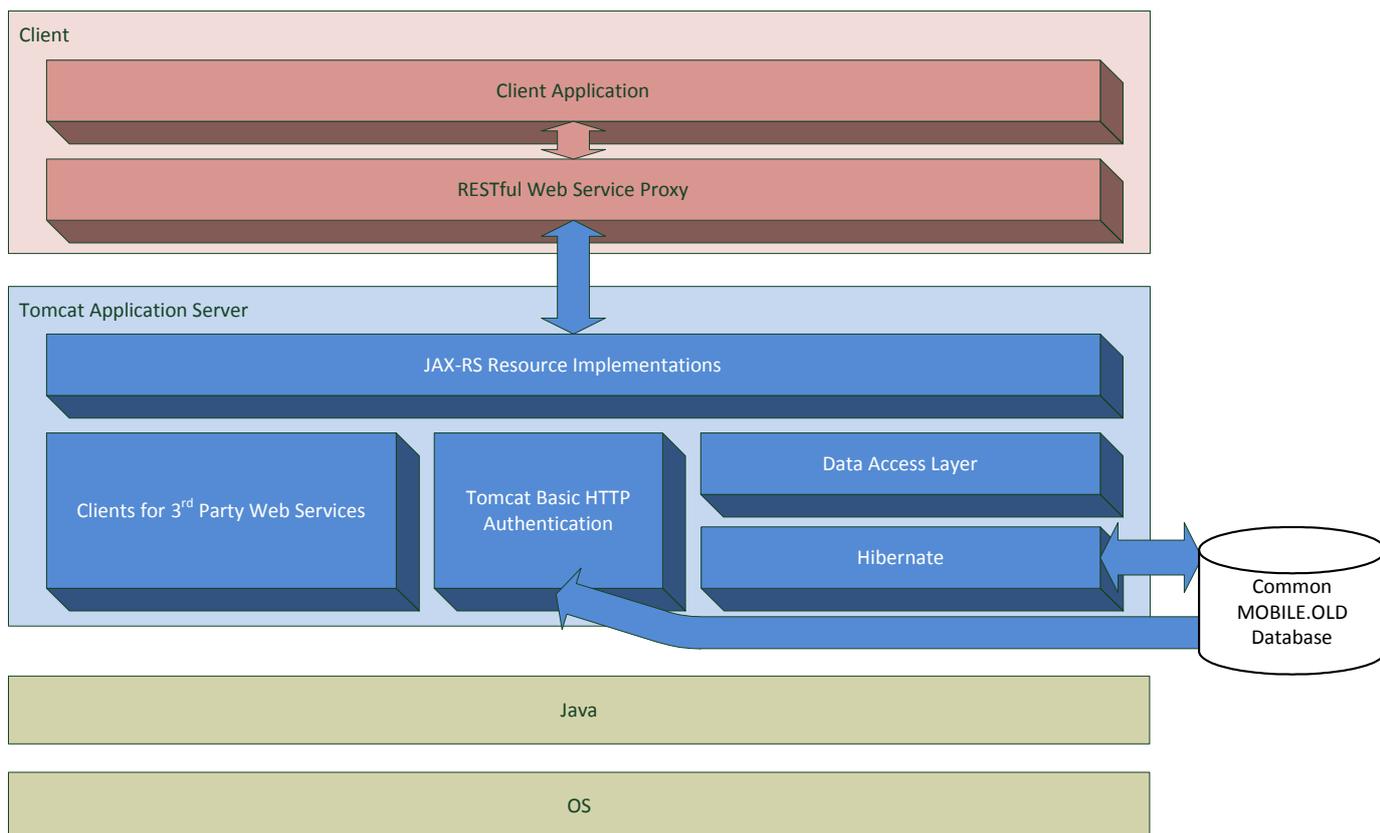
## 6.4 WEB SERVICE TECHNOLOGY

When mobile devices come into play, efficiency considerations gain importance. Transmitting data over wireless networks increases battery drain because power-hungry radio equipment is used. Also, transmitting higher amounts of data costs more money in telecommunication fees. These are two motivations to keep the transmitted data as small as possible. Additionally, complex data formats,

such as XML, consume significant amounts of CPU time, which leads to additional power consumption.

To accommodate the requirement for power preservation, a separate web service is being developed, which integrates with the previously described MOBILE.OLD web service on the database level. This web service is RESTful, which eliminates the overhead of a SOAP envelope by exploiting the HTTP protocol's possibilities (especially content type negotiation and authentication). For structuring data, the lightweight JSON format is used. It requires little overhead for structuring, and can be parsed fairly efficiently in native Android and JavaScript environments.

In order to determine the most suitable approach to develop such a web service, some representative technologies were evaluated by creating sample web services and testing them in terms of performance, resource efficiency, and difficulty of development. The compiled test results and a more detailed conclusion drawn from them can be found in Annex A. Summarized, Java-based solutions not only rank highest in performance, they also appear to be a good choice because the majority of developers is experienced on the Java platform, minimizing additional training efforts. JAX-RS, which is an established standardized API for implementing RESTful web services in Java, has been deemed the most suitable technology in this situation.



**Figure 60: Separate RESTful web service architecture**

While this web service will use its own database in the development phase, it will be adapted to use the common MOBILE.OLD database for interoperability with the other MOBILE.OLD web services. Thanks to **Hibernate**, an object-relational database mapper, switching between different databases is a matter of replacing a few lines of configuration and the JDBC driver. The **Data Access Layer**

introduces an additional abstraction layer, so that changes to the database schema don't affect the logic of the resource implementations.

The resources may not only use the database to acquire data, but also **3<sup>rd</sup> party web services**. The advantage to querying those web services directly from the client device is that the server can process the data so the client doesn't have to, reducing CPU- and power consumption on the client side. If such processing is not necessary, querying the web service directly from the client would be the better choice due to lower latency.

The web service uses **Tomcat's Basic HTTP Authentication** module to obtain a user's identity, which is used to determine, if they have the privilege to perform the desired operation on a certain resource, based on the roles assigned to the user. Since Basic HTTP Authentication is used, it is important that the production server will use HTTPS, so the user credentials aren't sent over the network in plain text.

The **resource implementations** contain the web service logic. Their job is to verify user permissions for certain operations, validate user input, and transform data. A class on this layer is roughly equivalent to one resource, which is represented by a URL. A subset of HTTP methods can be used to inspect and transform the resource.

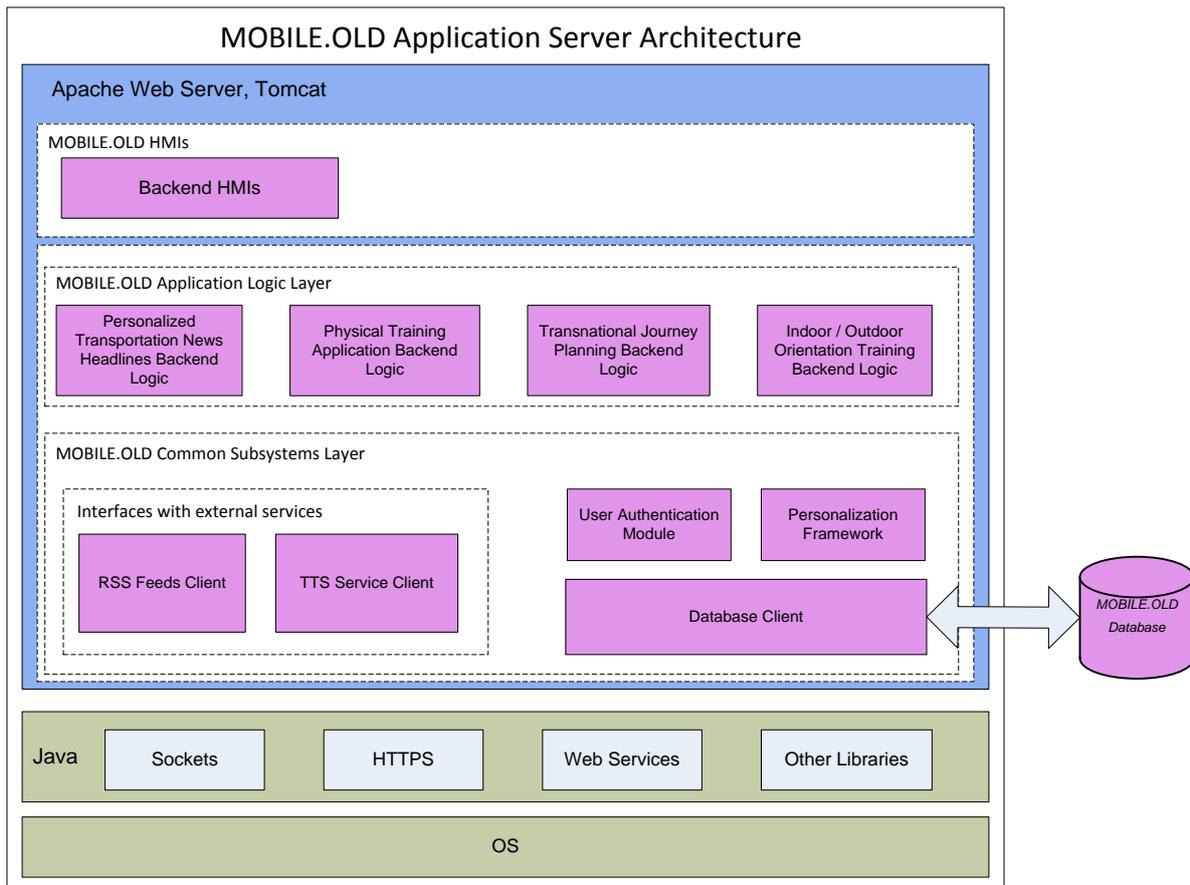
The **client application** does not interact directly with the web service, but does so through a **web service proxy**, which allows accessing resources on a high level, where manually performing HTTP requests is not necessary.

## 7. MOBILE.OLD APPLICATION SERVER ARCHITECTURE AND SMART PHONE ARCHITECTURE

### 7.1 APPLICATION SERVER ARCHITECTURE OVERVIEW

The MOBILE.OLD Application Server consists of a number of different components/modules. Specifically, it consists of the following top-level components as shown in the following figure:

- The MOBILE.OLD backend HMIs which are the administration web pages of the platform's services.
- The application logic components which implement the backend functionality of the MOBILE.OLD services.
- The interfaces with external services components which include the interfaces with the transportation news websites (via the RSS Feeds Client component) and the TTS service. The content will be up to date only if it is retrieved when the user requests it. Configurable content sources will allow the provision of targeted content for each user.
- The personalization framework component which stores the personalization preferences of each service.
- The User Authentication module which provides user identification and authentication functionality for the platform's services.
- The database client component which provides access to the MOBILE.OLD database.



**Figure 61: MOBILE.OLD Application Server Architecture**

As described above, the MOBILE.OLD application server is composed of the following components:

- The Backend HMIs which include the HMIs that are used by the administrator of the system to configure and monitor the services.
- The services application backend logic components, which implement the Personalized Transportation News Headlines backend news retrieval functionality, the Physical Training video hosting functionality, the Transnational Journey Planning and the Indoor / Outdoor Orientation Training backend functionality respectively.
- The RSS Feeds Client module, which is responsible for retrieving the transportation news content from the news websites.
- The TTS Service Client, which is responsible for connecting to the 3<sup>rd</sup> party TTS service and retrieving the audio that will be played in the user endpoint.
- The Personalization Framework, which is accessed by each service to retrieve and store the user's personalization preferences.
- The User Authentication Module, which provides user identification and authentication to the portal's services.

The Application Server will be implemented on top of the Apache Tomcat web server, an open source http server available on both Linux and Windows platforms. The MOBILE.OLD will use a MySQL database, hosted by a respective MySQL server.

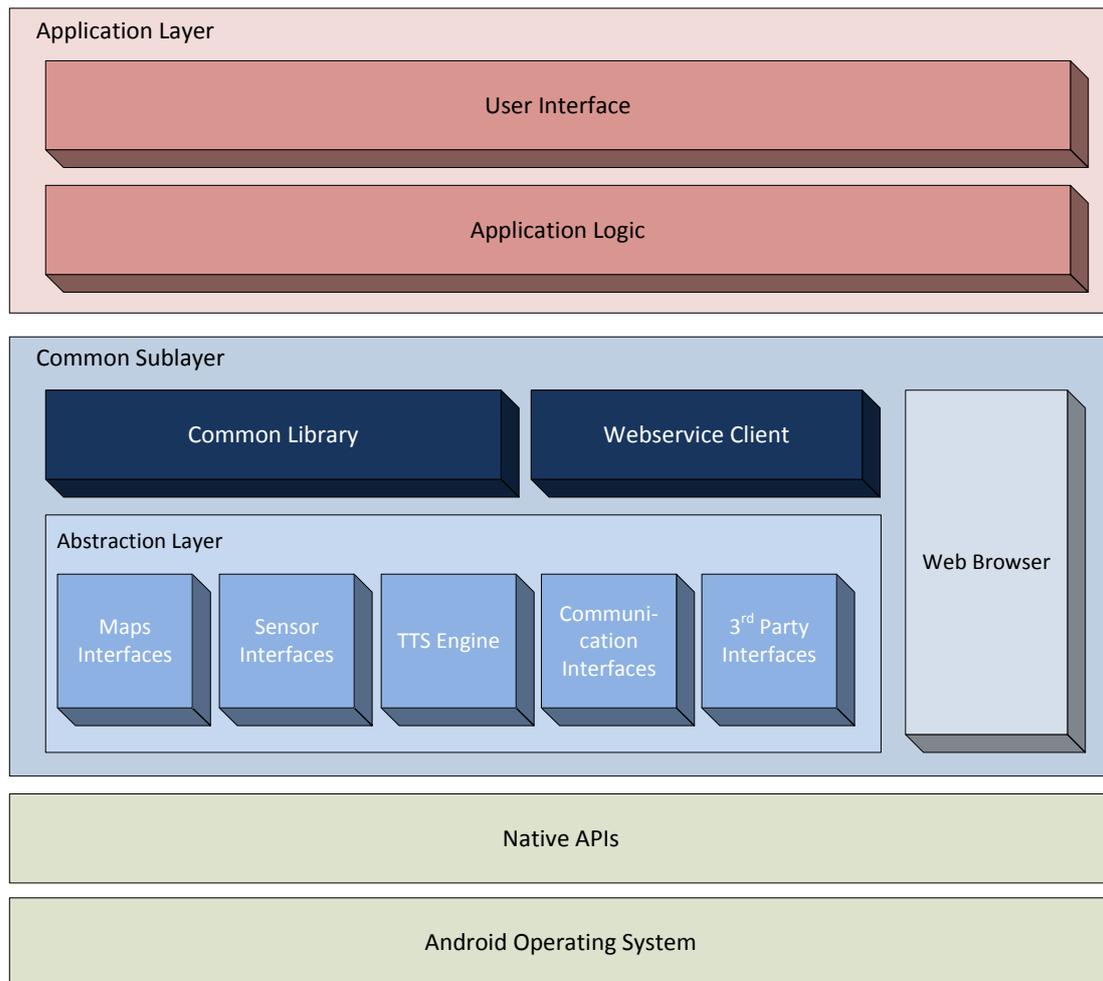
The nature of the MOBILE.OLD services requires that some user data shall be stored on the Service Provider side. This shall be done following the written agreement (informed consent) of the potential users that are subscribed to the services. The transmission of these data over the public Internet or a private IP network is performed over secure communication technologies. All communication interfaces are based on Internet technologies supporting standard and well established security mechanisms including SSL, HTTPS and Secure Web Services.

In order to further protect the user data at storage level, these are stored in the database and are related to a random user ID and not his/her real details. Even the interfaces to external systems (the web services) never include user identification data (such as their full names), just their IDs. They are only known to a Service Provider where a user/customer/account is created and are only handled by the administrator who creates the user account.

## 7.2 SMART PHONE SOFTWARE ARCHITECTURE

The architecture of the Smart Phone builds on the Android operating system and its native APIs. It includes libraries and components for abstracting the devices hardware and for interfacing with the MOBILE.OLD application server and web services in general. The architecture consists of the following main components:

- The **Application Layer** contains the business logic and user interface of the application. The application layer changes with every service implemented.
- In order to provide common base functionality and a basic abstraction from the Android System and APIs, the **Common Sublayer** is introduced.
- The **Native APIs** on top of the **Android Operating System** are the core the applications are built on.



**Figure 62: Architecture Overview of the Smart Phone**

The four main layers can be divided into sub-components as shown in Figure 41. These components can be described as following. This is done from bottom to top.

- The **Android Operating System** and the **Native APIs** providing a first hardware abstraction and interfaces to build on.
- The Common Sublayer includes an **Abstraction Layer**, that's main purpose is to simplify API access. It consists of five components:
  - The **Maps Interface** provides all necessary functionality for showing and interacting with maps in Android.
  - Relevant components are the **Sensor Interfaces** for interacting with the Android GPS wrappers and coarse orientation sensor as well as the accelerometer and magnetic field sensors in order to determine the devices exact orientation. Any additional sensor interaction is also done via this interface.
  - The **3<sup>rd</sup> Party Interfaces** contain all external logic that is based on the Android operation system and the native APIs.
  - **Communication Interfaces** include the necessary communication interfaces and technologies for networking and sending SMS.

- The TTS Engine can be used for fast Text-to-Speech output on the device.
- Next to the **Web Browser**, the core component of the architecture is the **Common Library**. It helps to maintain a common base for different applications. The library builds on the abstraction layer and contains components for common application logic and user interface modules.
- The **Webservice Client** acts as proxy client and connects the application with the application server.
- The **Application Logic** processes data from the Common Library and the Webservice Client and updates the UI.
- The **User Interface** is the visible part of the application.

## 8. DEVELOPMENT PLAN

The following table depicts the development plan of the MOBILE.OLD system components.

Item	Responsible	Version	Delivery Date	Comment
Personalized Transportation News Headlines Service	SERPO	1.0	30/06/13	N/A
Geofencing Service	SMNS	1.0	30/06/13	N/A
Physical Training Service	BLPT	1.0	30/06/13	N/A
Transnational Journey Planning Service	SFV	1.0	30/06/13	N/A
Hiking Application	SEPRO	1.0	30/06/13	N/A
Orientation and Geotag Application	UoAS	1.0	30/06/13	N/A
Indoor / Outdoor Orientation Training Application	SEPRO	1.0	30/06/13	N/A
First Aid Application	UoAS	1.0	30/06/13	N/A
Checklist Application	UoAS	1.0	30/06/13	N/A
Orientation and Mobility Games	UoAS	1.0	30/06/13	N/A
MOBILE.OLD Database	SEPRO	1.0	30/06/13	N/A
MOBILE.OLD Personalization Framework	SEPRO	1.0	30/06/13	N/A
MOBILE.OLD Interfaces with External Services	SEPRO	1.0	30/06/13	N/A
Personalized Transportation News Headlines Service	SERPO	2.0	31/12/13	N/A
Geofencing Service	SMNS	2.0	31/12/13	N/A
Physical Training Service	BLPT	2.0	31/12/13	N/A

Transnational Journey Planning Service	SFV	2.0	31/12/13	N/A
Hiking Application	SEPRO	2.0	31/12/13	N/A
Orientation and Geotag Application	UoAS	2.0	31/12/13	N/A
Indoor / Outdoor Orientation Training Application	SEPRO	2.0	31/12/13	N/A
First Aid Application	UoAS	2.0	31/12/13	N/A
Checklist Application	UoAS	2.0	31/12/13	N/A
Orientation and Mobility Games	UoAS	2.0	31/12/13	N/A
MOBILE.OLD Database	SEPRO	2.0	31/12/13	N/A
MOBILE.OLD Personalization Framework	SEPRO	2.0	31/12/13	N/A
MOBILE.OLD Interfaces with External Services	SEPRO	2.0	31/12/13	N/A

## 9. EVALUATION AND DEPLOYMENT PLAN

This section gives an overview of the evaluation and deployment of the MOBILE.OLD services in each pilot country with the involvement of seniors. First of all the evaluation plan will be described. The evaluation plan consists of a common approach that will be implemented in all pilot countries in order to provide for a set of homogenous feedback for service optimization.

Secondly the section will describe the deployment plan to perform the trial sessions in the pilot countries. Pilot specific issues will be taken into account and discussed here separately.

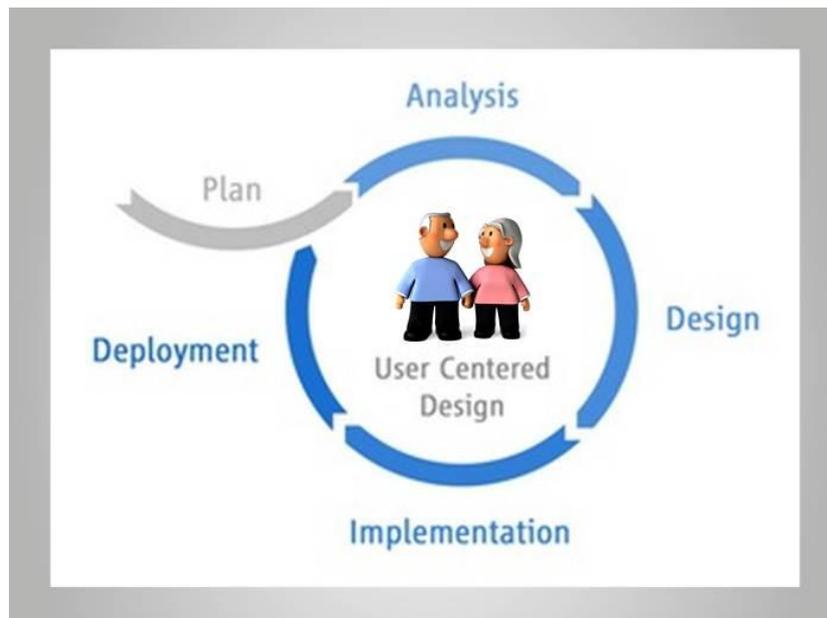


Figure 63 - MOBILE.OLD User Centered Design

### 9.1 SERVICES EVALUATION PLAN

The evaluation plan is following an iterative approach, where several phases are implemented to involve the end-user in the design of the services, the evaluation of the usability and the MOBILE.OLD validation.

The usability study will use a hybrid approach that combines both studies in a controlled environment (at the organization premises) as well as field studies (on the road and at the user's home). Different methods are used for investigating different usability issues and gathering data.

The efforts will focus on evaluating components of the mobile and TV interface. In a controlled environment, the research will focus on interface layout, information presentation schemes, design of menu and link structures and data entry methods.

In field studies usability testing will focus on the services with application performance issues, user interaction, user behaviour and attitude towards mobile applications.

The usability evaluations will be made in three iterative phases. The results of these evaluations will be presented to the technical partners for development. The first phase will be a Mock-Up phase. Phase 2, 3 and 4 will be conducted as part of the pilot to evaluate the prototypes. The current document will give a description of the activities during the Mock-Up phase. The first, second and the

third prototype evaluation phases will be briefly introduced here. The methodology and planning of these phases is further detailed in D5.1 as part of the MOBILE.OLD trial set-up.

Finally the evaluations will also test the services with the use of tablets. Different usability studies have already demonstrated how tablets can lower some of the barriers for internet based services, due to its non-technical look and feel accompanied by the simple touch based navigation.<sup>1</sup> More and more seniors are becoming familiar with tablets. Although this is not in the main objectives of the MOBILE.OLD project, it is considered that the user interaction on tablets is essential to take into account for the future exploitation of MOBILE.OLD.

### 9.1.1 MOCK-UP EVALUATION

At the Mock-Up evaluation a series of mock-ups will be available, that will be presented to the users on the TV and on the Smart Phone at the premises of the end-user organization. With the help of these mock-ups a usability evaluation will take place on the first design. Data at this stage will be collected through observation, an interview survey and a verbal protocol. The session will be held with 5 users and one caregiver (or at least one person with caregiver background) in each pilot in a controlled environment during approximately 1.5 hour.

#### The following actions will take place during the sessions:

- **Interface layout:**

On each mock-up, users will be asked to give their opinion on the look of the services, focusing on aspects such as colour, contrast, places for buttons and information.

- **Information presentation schemes:**

On each mock-up, users will be asked to perform a certain task that will demonstrate whether the buttons are easy to find on the screen and whether they are sufficiently self-explanatory. The researcher will write down notes and comments.

- **Design of the menu and link structure:**

The structure of the different MOBILE.OLD services in a single application will be pre-defined by the technical partners. This structure will then be tested with the end-users through a card sorting method.

A board will be made with predefine cards that represent the navigation structure. The predefined cards will contain the taxonomy of the menu in the different languages of the pilot countries and a photo or picture that represents the content in an image.

The users will be able to give their feedback on the taxonomy and the structure. The user can leave the structure as it is and comment on it, or move the cards around to change the structure. By comparing the visual image and the taxonomy on the card, the user can give comments on the words used.

This will give clear feedback on the organization of the menu and optimize the results for a powerful user experience. It will make clear whether the taxonomy used is representing the content of that section well enough.

---

<sup>1</sup> K.Werner, F. Werner, 'Tablets for seniors: Bridging the digital divide' *Gerontechnology* 2012, 11 (2), 208.

### 9.1.2 PRE PROTOTYPE EVALUATION

Within the scope of the user centred design approach a pre prototype evaluation will be conducted in order to provide early feedback to the technical partners for their further activities. The pre prototypes will be of basic functionality and just an intermediate result of the services implementation process. The session will be held with 3-5 users in each pilot and one caregiver (or at least one person with caregiver background) in a controlled environment and in the field, during approximately 1.5 hour. The research methodology followed will be the same as for the first and second prototype evaluation.

**Mobile interaction and user behaviour:** In a field test, users will be asked to use the orientation application to find a nearby location and take a photo of this location. While using it, users are asked to think out loud and their comments are recorded. The pilot partners will prepare these actions with the help of short guidelines and interview questions that can be used as support for the researchers during the sessions.

### 9.1.3 FIRST PROTOTYPE EVALUATION

The early prototype testing will be held in a pilot, where 20-25 end-users and one caregiver (or at least one person with caregiver background) are using the complete prototype with the Smart TV and the Smart Phone at the premises of the end-user organizations. The early prototype evaluation will be therefore more similar to a laboratory study. The users take part in usability tests in a controlled setting. The advantage is that users will use the system in an interruption free environment. A limitation of classical usability testing is that it only assesses the first encounter. This doesn't take into account that observed incidences may occur by mere unfamiliarity. To separate these incidences from the true problems MOBILE.OLD will conduct a repeated measures study, and track which problems disappear after repeated use.

The partners will use different methodologies for the evaluation, in order to provide for consistent feedback to the technical partners. The following is a short introduction of the methodologies that will be considered and worked out as part of deliverable 5.1.

#### Research Methodology:

- **Interviews:**

Interviews can be structured, semi-structured or unstructured. With structured interviews quantitative data can be collected, with the latter two, more qualitative data is collected.

- **Observation:**

Software will be used to film the participant but also the mobile screen, showing which actions are performed in the application. The video footage can then be easily reviewed to detect incidents. These incidences will then be analysed to see if they occur with multiple users, generating a design change which is a suggestion to prevent the incident in a specific task.

- **Think Aloud:**

For the current study, we will use the think aloud procedure to capture qualitative research data. This information will be used to identify usability problems which need to be addressed by the developers.

- **Performance Measures:**

As part of a task analysis quantitative data can be collected such as response time, execution time, dual task performance.

- **Questionnaire:**

For this study, the ASQ and PSSUQ questionnaires are used to measure different aspects of usability. These surveys have been validated and are free to use.

The PSSUQ is to be administered after the participants have completed all tasks. This questionnaire is an overall measurement of all the services.

Additionally, questions will be added to the questionnaire that relate to specifications in the relevant deliverable.

**Repeated measures study:**

A repeated measures usability testing method will be used to see how the learning curve of the participants evolves and what kind of incidences remain after the participating seniors are more familiar with the interfaces. The participants will be given a number of tasks they have to perform using the applications of the MOBILE.OLD project. To separate the errors that are caused by a faulty or inconvenient design from the ones that are caused by novelty, the Dutch pilot will use a repeated measures method. After measuring multiple times, the pilot will demonstrate which problems disappear with repeated use. The problems that remain are the ones that are most important to solve in a new design and prototype version. As this method wants to show a learning effect, it is necessary that the tests are done with the same elderly, to be able to compare their first results with their other (newer) results. For this study a sample of 20-25 participants will be used.

#### **9.1.4 SECOND PROTOTYPE EVALUATION**

In the second prototype evaluation the 10-15 users and one caregiver (or at least one person with caregiver background) per pilot site are observed using the MOBILE.OLD system during a longer time frame in their own environment.

A lack of control may make it more difficult to evaluate the system, but on the other hand the evaluation will be focusing on a more natural surrounding and use. Therefore the second prototype evaluation will include the use of log files, making it possible to monitor the amount of use, the interaction, mouse clicks, hovering, response time, etc. Additionally the evaluation will use the research methodologies described in the section above.

## **9.2 DEPLOYMENT PLAN**

The deployment plan will explain all the actions required for the setup of the pilots. This will be used as a basis for the setup of the trials in the pilots.

### **9.2.1 USER SELECTION**

Within the scope of the MOBILE.OLD project active seniors (aged 60+) who like to maintain their mobility level by using novel ICT solutions are selected for the evaluation phases.

The selected users could also include the users who were participating within the user requirements capturing phase.

No restrictions are placed on the repetition of involving the same users in different phases. Therefore users that have participated in the definition of the user requirements can also be included in the prototype evaluations. Additionally users from the pre-prototype evaluation and prototype evaluations can be the same.

### 9.2.2 USER TRAINING

The users receive a short training regarding the interaction with the Smart TV and the Smart Phone prior to the evaluation activities. This includes:

- **Smart Phone usage**
  - Make a call and write a SMS
  - To make and view a photo
  - To start and close an App
  
- **Smart TV (Full HD TV + Android TV Box) usage**
  - To start and close an App
  - To start the browser and view a website

Furthermore a short tutorial on the MOBILE.OLD services will be provided by the technical partners. The elderly will be trained by the pilots on the basis of this tutorial prior to the evaluations.

### 9.2.3 EQUIPMENT ACQUIREMENT

The pilot site partners use a 5 inch android Smart Phone for the evaluation activities referring to the user requirements. Every pilot site partners should buy the same equipment in order to obtain homogenous results.

The current price for a new Samsung Galaxy Note I Smart Phone is 379€ (new device), 320€ (second hand) on amazon.com. The MOBILE.OLD services will initially target TV platforms and over the top boxes that are based on Android.

As far as Tablets are concerned the **Google Nexus 7 and 10** will be tested during the pre-prototype and the 1<sup>st</sup> Prototype evaluation activities. Based on these results a decision is made on which tablet is going to be used for the 2<sup>nd</sup> trial phase.

In terms of price-performance ratio, it was decided among the consortium to use the **MINIX Neo X5** TV Box, which is described in Table 1.

The Spanish pilot site partner will equip the test-households with the **SAFEVIEW IPO-85X Android** Set-Top Box which features a 2500MIPS processor, full HD H.264 decoding, WiFi, Android 4 and a WebKit based web browser.

#### 9.2.3.1 EQUIPMENT NEEDED:

- for the **Mock-Up Evaluation**: 1x Full HD TV, PC with PDF Viewer, Smart Phone

- for the **Pre-Prototype Evaluation**: 1x Minix Neo X5, Full HD TV, Smart Phone, AirMouse/Keyboard, Google Nexus 7, Google Nexus 10
- for the **1<sup>st</sup> Prototype Evaluation**: 1x Minix Neo X5, HD TV, Smart Phone, AirMouse/Keyboard, Google Nexus 7, Google Nexus 10
- for the **2<sup>nd</sup> Prototype Evaluation**: up to 15 Minix Neo X5, HD TV, Smart Phone, AirMouse/Keyboard, Google Nexus 7 or Google Nexus 10 (decision depends on feedback collected during prior evaluation phases)

## 9.2.4 CONTRACT FOR THE USAGE OF THE EQUIPMENT

Under this section the contract for the usage of the equipment which is presented to the elderly within the start of the second prototype evaluations added by every pilot site partner.

- **Austrian Pilot Site**

### **Leihvereinbarung**

#### **Projekt: MOBILE.OLD**

Sehr geehrter XY

Im Rahmen der Testphase des MOBILE.OLD Projekts erhalten Sie für den Zeitraum **XX. X 2013 bis XX. X 2013** unentgeltlich folgende Geräte:

- **1 Stück Flachbildfernseher, Marke:..., SN:...  
inklusive Kabel, Fernbedienung und Handbuch**
- **1 Stück Mobiltelefon, Marke:..., SN:...  
inklusive Gerätetasche, Ladegerät und Handbuch**

Die Leihgeräte sind nach dem Beenden der Testphase wieder mit allen zur Verfügung gestellten Zusatzausstattungen an die LIFEtool gemeinnützige GmbH zurück zu geben.

Für Verlust und Beschädigungen während der Leihdauer haften Sie.

Die Leihgeräte werden vor und nach der Leihstellung von LIFEtool auf ihre Funktionstüchtigkeit geprüft.

Bei Mängel, Problemen oder technischen Schwierigkeiten, bitten wir Sie um sofortige Verständigung.

Mit Ihrer Unterschrift bestätigen Sie die Übernahme der oben genannten Geräte und dass Sie die o.a. Leihbedingungen zur Kenntnis genommen haben.

\_\_\_\_\_  
Datum

\_\_\_\_\_  
Unterschrift

- **Dutch Pilot Site**

### **Bruikleenovereenkomst**

#### **MOBILE.OLD**

Het **Nationaal Ouderenfonds**, gaat een bruikleenovereenkomst aan met

**Mevrouw/Meneer ....**

**Adres:**

verder te noemen de **gebruiker**.

Onder de volgende voorwaarden:

#### **1. Algemeen**

In het raamwerk van het MOBILE.OLD project stelt het Nationaal Ouderenfonds apparatuur ter beschikking ten behoeve van een test fase. De apparatuur is en blijft eigendom van het Nationaal Ouderenfonds en wordt aan de gebruiker in bruikleen gegeven.

#### **2. Apparatuur**

Het Ouderenfonds stelt de volgende apparatuur ter beschikking voor de periode van datum ondertekening van deze overeenkomst tot het einde van de testfase.

- Flatscreen TV, inclusief kabels, afstandsbediening en gebruiksaanwijzing van het merk....
- Smart Phone, inclusief kabels, oplader, gebruiksaanwijzing en pre-paid sim kaart van het merk ....

#### **3. Gebruik**

- De gebruiker stelt de apparatuur niet aan derden ter beschikking
- Het Nationaal Ouderenfonds is verantwoordelijk voor het leveren en installeren van de apparatuur.
- De gebruiker is verantwoordelijk voor het in goede staat, en operationeel houden van de apparatuur. Hij neemt de nodige zorgvuldigheid in acht ter voorkoming van diefstal, verlies of beschadiging van de apparatuur.
- De gebruiker meldt verlies, diefstal, beschadiging of gebreken van de apparatuur zo snel mogelijk aan de telefonische helpdesk, zodat deze aangift kan doen bij de politie en het nummer laat blokkeren.
- De gebruiker wordt binnen de aangegeven kantooruren, bijgestaan door de telefonische helpdesk en het Nationaal Ouderenfonds .

#### **4. Aansprakelijkheid**

Het Nationaal Ouderenfonds is niet aansprakelijk voor schade of boetes als gevolg van het gebruik van de apparatuur.

#### **5. Beëindiging overeenkomst**

Het Nationaal Ouderenfonds behoudt zich te alle tijde het recht voor het gebruik te beëindigen. Indien de testfase beëindigd is, wordt de apparatuur opgehaald door het Nationaal Ouderenfonds.

**Aldus overeengekomen en in tweevoud opgemaakt te Bunnik door:**

#### **Nationaal Ouderenfonds**

Naam en handtekening

Mevr./Menr.

Naam en handtekening

- **Centrul Pilot din Romania**  
**Proiect: MOBILE.OLD**

### **Contract de imprumut**

Stimate domnule / Stimate doamna.....,

In cadrul fazei de testare a proiectului MOBILE.OLD, in perioada XX. X și 2013 XX. X 2013 va vor fi puse la dispozitie, in regim de imprumut fără taxă, urmatoarele dispozitive:

- 1 TV cu ecran plat, marca: .....Nr. Serie: .....  
Cablul, telecomandă și manuale de utilizare incluse
- 1 telefon mobil, marca: .....Nr. Serie: .....  
Husă de transport, incarcator si manual de utilizare incluse.

Va facem cunoscut faptul ca, după finalizarea fazei de testare, dispozitivele si accesoriile acestora mentionate mai sus, vor trebui înapoiate centrului care vi le-a furnizat in regim de imprumut, si anume, Fundatiei Ana Aslan International.

Veti fi considerat răspunzător in cazul pierderii sau deteriorarii dispozitivelor pe perioada in care acestea se vor afla in posesia dvs. in regim de imprumut.

Pentru a se constata buna functionare a dispozitivelor, acestea vor fi testate de catre Fundatia Ana Aslan International si SIEMENS Romania, înainte și după serviciul de împrumut. Daca ati sesizat vreo problema tehnica la unul dintre dispozitive, va rugam sa ne anuntati in cel mai scurt timp.

*Prin semnătura dumneavoastră confirmați* că ati citit, ati inteles si ati acceptat conditiile de imprumut mentionate in prezentul contract.

Data

Semnatura

- **Spanish Pilot site**

**ACUERDO DE PRÉSTAMO****Proyecto: MOBILE.OLD**

En calidad de usuario D. \_\_\_\_\_, recibo como parte de la fase de prueba del proyecto MOBILE.OLD recibo sin cargo con motivo de mi participación en el proyecto Mobile.Old, para el período XX. X y XX de 2013. X 2013 los siguientes dispositivos:

- 1 TV de pantalla plana, marca \_\_\_\_\_, modelo \_\_\_\_\_ - y número de serie \_\_\_\_\_ que incluye cable, control remoto y manual de instrucciones.
- 1 teléfono móvil de marca \_\_\_\_\_, modelo \_\_\_\_\_ y número de serie: \_\_\_\_\_, que Incluye la realización manual de la caja, el cargador y el

Con las siguientes condiciones:

- Las unidades deberán ser devueltas después de completar la fase de prueba de nuevo con todos sus aparatos adicionales.
- En caso de pérdida y daños durante el período de alquiler, el usuario será responsable de su restitución.  
Las unidades que se ceden en préstamo han sido probadas antes del servicio de préstamo y serán provadas a la entrega del mismo para verificar su correcto funcionamiento.
- El usuario es responsable de ello en buen estado, y mantener el equipo. Él toma el cuidado necesario para evitar el robo, pérdida o daños en el equipo.
- El usuario deberá reportar inmediatamente la pérdida, robo, daño o defectos del equipo tan pronto como sea posible.
- El prestador de los dispositivos se reserva el derecho en cualquier momento de retirar los dispositivos al usuario aunque la fase de prueba no se hubiera finalizado.

Con la firma del presente documento acepto el préstamo de los dispositivos mencionados arriba en las condiciones recogidas en este documento.

\_\_\_\_\_  
Firma Fecha

## 9.2.5 HELPDESK ESTABLISHMENT

- **On the Austrian Pilot Site**



**Stefan Schürz BSc**  
+43664/ 235 17 43  
stefan.schuerz@lifetool.at

Hours of Business:  
Mo-Do: 10:00-16:00  
Fr: 10:00-14:00



**DI Alfred Doppler**  
+43732/ 99 70 56  
alfred.doppler@lifetool.at

Hours of Business:  
Mo-Do: 09:00-15:00  
Fr: 09:00-13:00

- **On the Dutch Pilot Site**



**Nina van der Vaart**  
030-6567774  
[n.vandervaart@ouderenfonds.nl](mailto:n.vandervaart@ouderenfonds.nl)

Hours of Business:  
Mo-Fr: 10:00-16:00



**Gerard van Loon**

030-6567774

[g.vanloon@ouderenfonds.nl](mailto:g.vanloon@ouderenfonds.nl)

Hours of Business:

Mo-Fr: 10.00-16.00

- **On the Romanian Pilot Site**



**Luiza Spiru, MD, PhD**

+4021 312 46 96

lsaslan@brainaging.ro

Hours of Business :

Mo-Fr: 10:00-16:00



**Ileana Turcu**

+40723 18 66 35

it@brainaging.ro

Hours of Business :

Mo-Fr: 10:00-16:00

- **On the Spanish Pilot Site**



**Marta Blanco Johnson**

+34 674 73 49 35

Hours of Business:

Mo-Do: 10:00-18:00

Fr: 10:00-14:00



**Montserrat Roura Portillo**

+34 637 498 464

mobile\_old\_r@madridnetwork.org

Hours of Business:

Mo-Do: 9:00-18:00

Fr: 10:00-14:00



**José Manuel Laperal González**

+34 699936761

mobile\_old\_l@madridnetwork.org

Hours of Business:

Mo-Do: 9:00-18:00

Fr: 10:00-14:00

## 9.2.6 INSTALLATION OF THE MOBILE.OLD PROTOTYPES

### 9.2.6.1 CENTRALLY IN THE PREMISES OF EVERY PILOT SITE

- **On the Austrian Pilot Site**

LIFEtool's Consultation Room



- Wireless as well as cable-based internet connection (high bandwidth)
- Seats for 20 people
- Suitable for wheelchairs
- Presentation equipment (beamer, smart board, flipchart, computers)
- Cozy atmosphere

- **On the Dutch Pilot Site**

The National Foundation for the Elderly has at its premises a room for focus group meetings and tests. At the activity centres the NFE can use different computer rooms as well as meeting rooms for tests and discussions. The picture below is the consultation room at the NFE premises, which is suitable for wheelchairs and has the entire infrastructure required for tests (internet, computers, etc.).



- **On the Romanian Pilot Site**

Ana Aslan International Foundation holds the Clinic for Memory Diseases, which is also the FAAI premise for end-users testing related to platforms evaluation in AAL projects. The consultation room has the suitable endowment for such activities:

- Wireless as well as cable-based internet connection (high bandwidth)
- Seats for 25 people
- Suitable for wheelchairs
- Presentation equipment (beamer, smart board, flipchart, computers)
- Cozy atmosphere



A multidisciplinary team of doctors, nurses, and researchers, as well as IT&C assistants, well trained in previous AAL projects, is also available.

- **On the Spanish Pilot Site**

Elder Center installations in Getafe pilot site:



- Located in the center of city, closed to the elder citizens.
- Training rooms with equipment for presentations (projector, smart board, flipchart, and computers).
- Auditorium/press room for groups presentations .
- Private rooms for meetings.
- Connection with social services.
- Wireless as well as cable-based internet connection (high bandwidth).

#### 9.2.6.2 INSTALLATION AT USER'S HOUSEHOLDS

The Android TV-Box can be easily installed at the user's households. No extensive installation activities are required. The internet connection can be realized with a cable based connection (LAN), a 3G Dongle/Router or a wireless router (WLAN).

## 10. SERVICES VALIDATION PLAN

### 10.1 PERSONALIZED TRANSPORTATION NEWS HEADLINES SERVICE

#### 10.1.1 TESTS CONFIGURATION

The Personalized Transportation News Headlines service will be tested using an Android STB for the TV endpoint. For the Smart Phone endpoint of the mobile environment, the service will be tested using a Samsung Galaxy S Smart Phone device.

#### 10.1.2 VALIDATION SCENARIOS

Test Case Id	MDO_PTNH_TEST_001		Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
To verify that the users can view transportation news and weather information from the TV and mobile endpoint.	1	The user opens the MOBILE.OLD application and selects the Personalized Transportation News Headlines service.	The Personalized Transportation News Headlines service main screen is displayed.
	2	The user selects the "Ground Transportation" option from the news categories menu.	The news screen displays regional, national and European transportation news.
	3	The user selects the "Traffic Info" option from the news categories menu.	The news screen displays regional, national and European traffic information news.
	4	The user selects the "Air Transportation" option from the news categories menu.	The news screen displays regional, national and European air travel information news.
	5	The user selects the "Weather info" option from the news categories menu.	The news screen displays regional, national and European weather information.
	6	The user selects the "Strikes info" option from the news categories menu.	The news screen displays regional, national and European strikes information news.

Test Case Id	MDO_PTNH_TEST_002		Comments
Test Purpose	Test Case Description		Test Case Pass Criteria

Test Case Id		MDO_PTNH_TEST_002	Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
To verify that the users can view transportation news for the destination of the journey that they have planned using the TJP service.	1	The user opens the MOBILE.OLD application and selects the Personalized Transportation News Headlines service.	The Personalized Transportation News Headlines service main screen is displayed.
	2	The user selects the “Go to TJP service” option from the journey planning menu.	The TJP service opens.
	3	The user plans a journey using the TJP service and returns to the PTNH service.	The user has planned a journey and is back to the PTNH service.
	4	The user selects the “Destination News” option from the journey planning menu.	The news screen displays transportation information about the user’s destination

Test Case Id		MDO_PTNH_TEST_003	Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
To verify that the users can configure the service’s news area level (regional, national, European).	1	The user opens the MOBILE.OLD application and selects the Personalized Transportation News Headlines service.	The Personalized Transportation News Headlines service main screen is displayed.
	2	The user presses the settings button and is transferred to the service configuration screen.	The PTNH configuration screen is shown.
	3	The user unchecks all the area levels from the “Transportation News Areas” list, except the Regional setting, and presses the back button.	The service main screen is displayed.
	4	The user navigates to each news categories item.	Each news categories item displays only regional information.

Test Case Id		MDO_PTNH_TEST_003	Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
	5	The user navigates to the configuration screen and selects a combination of area levels from the “Transportation News Areas” list, and presses the back button.	The service main screen is displayed.
	6	The user navigates to each news categories item.	Each news categories item displays information for only the selected, from the configuration screen, areas.

Test Case Id		MDO_PTNH_TEST_004	Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
To verify that the users can configure the service’s provided categories of information.	1	The user opens the MOBILE.OLD application and selects the Personalized Transportation News Headlines service.	The Personalized Transportation News Headlines service main screen is displayed.
	2	The user presses the settings button and is transferred to the service’s configuration screen.	The PTNH configuration screen is displayed.
	3	The user selects a combination of news categories from the categories of information screen, and presses the back button.	The service’s main screen is displayed.
	4	The user navigates through the displayed news categories.	The displayed categories are the ones that the user has selected from the configuration screen.

## 10.2 GEOFENCING SERVICE

### 10.2.1 TESTS CONFIGURATION

The Geofencing service has the following main participating actors (see section 4.2): the elders and the elder caretakers of family. The elders use an Android based SAMSUNG Smart Phone, while the elder caretakers or the family must use:

- a computer connected to the Internet for *elders’ profile management* working mode;
- a computer or a smart TV for *elders’ monitoring* working mode;

- a mobile phone or a Smart Phone for receiving SMS messages.

### 10.2.2 VALIDATION SCENARIOS

Test Case Id	MDO_GEOFENCING_TEST_001		Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
Access the elders' profile management working mode.	1	The caretaker opens the MOBILE.OLD application and selects the Geofencing service.	The Geofencing service main menu is displayed.
	2	The caretaker logs in (using the username and password provided by the system administrator) into the elders' profile management menu.	The elders' profile management menu is displayed and the caretaker has access to the following functionalities: create/remove/edit the geofencing areas for each elder

Test Case Id	MDO_GEOFENCING_TEST_002		Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
Create a geofencing area for an elder (the caretaker has already accessed the elders' profile management menu).	1	The caretaker selects the elder.	On the screen is displayed a list that contains the geofencing areas defined for the selected elder. The caretaker can highlight on the map a geofencing area by selecting it from the list.
	2	The caretaker presses the <i>Add geofencing area</i> button.	The geofencing area definition menu is displayed.
	3	In the geofencing area definition menu the caretaker fills the <i>Geofencing area name</i> , the <i>Geofencing area type</i> (see section 4.2.1) and the geographic points which define the geofencing area. At the end, the caretaker presses the <i>Save</i> button.	The elders' profile management menu is displayed. The geofencing area list contains the new geofencing area and it is highlighted on the map.

Test Case Id	MDO_GEOFENCING_TEST_003	Comments

Test Purpose	Test Case Description		Test Case Pass Criteria
Edit a geofencing area for an elder (the caretaker has already accessed the elders' profile management menu).	1	The caretaker selects the elder.	On the screen is displayed a list that contains the geofencing areas defined for the selected elder. The caretaker can highlight on the map a geofencing area by selecting it from the list.
	2	The caretaker selects from the list a geofencing area and presses the <i>Edit</i> button.	The geofencing area definition menu is displayed and all the fields are filled with the corresponding data.
	3	In the geofencing area definition menu the caretaker edits the <i>Geofencing area name</i> , or the <i>Geofencing area type</i> (see section 4.2.1) or the geographic points which define the geofencing area. At the end, the caretaker presses the <i>Save</i> button.	The elders' profile management menu is displayed. The edited geofencing area is highlighted on the map.

Test Case Id	MDO_GEOFENCING_TEST_004		Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
Delete a geofencing area for an elder (the caretaker has already accessed the elders' profile management menu).	1	The caretaker selects the elder.	On the screen is displayed a list that contains the geofencing areas defined for the selected elder. The caretaker can highlight on the map a geofencing area by selecting it from the list.
	2	The caretaker selects from the list a geofencing area and presses the <i>Delete</i> button.	The selected geofencing area is no longer available in the displayed list.

Test Case Id	MDO_GEOFENCING_TEST_005		Comments
Test Purpose	Test Case Description		Test Case Pass Criteria

Test Case Id		MDO_GEOFENCING_TEST_005		Comments
Test Purpose	Test Case Description		Test Case Pass Criteria	
Monitor the elders' position.	1	The caretaker opens the MOBILE.OLD application and selects the Geofencing service.	The Geofencing service main menu is displayed.	
	2	The caretaker logs in (using the username and password provided by the system administrator) into the elders' monitoring menu.	The elders' monitoring menu is displayed. On the map included in the menu are highlighted the positions of all elders. The markers have two different colours to discriminate the following situations: elder inside the geofencing area and elder outside the geofencing area.	
	3	The caretaker moves the mouse above a marker which highlights the position of an elder.	After a short period of time a tooltip appears with more information about the selected elder.	
	4	The caretaker clicks on the marker.	On the map are highlighted the geofencing areas for the selected elder.	

Test Case Id		MDO_GEOFENCING_TEST_006		Comments
Test Purpose	Test Case Description		Test Case Pass Criteria	
Detection of deviations from a geofencing zone.	1	The elder starts on the Smart Phone the MOBILE.OLD application.	The Geofencing service main menu screen is displayed on the Smart Phone (it contains a map where are highlighted the position of the elder and the geofencing areas).	
	2	The elder starts outside activities.	The elder position is continuously monitored and it is updated on the map.	

Test Case Id	MDO_GEOFENCING_TEST_006		Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
	3	The older person leaves the geofencing area.	A warning is generated by the older person's Smart Phone (the phone starts to ring or/and vibrate and a notification message is displayed on the screen).
	4	The older person acknowledges the warning generated by the mobile phone.	The Geofencing service continues to monitor the elder position.
	5	The older person does not acknowledge the warning or he or she does not return to the geofencing area within a certain time interval (specified by the user profile).	A notification (SMS message) is sent to his or her family or to the caregiver.

## 10.3 PHYSICAL TRAINING SERVICE

### 10.3.1 TESTS CONFIGURATION

The Physical Training service will be tested using an Android STB for the TV endpoint. For the Smart Phone endpoint of the mobile environment, the service will be tested using a Samsung Galaxy S Smart Phone device.

### 10.3.2 VALIDATION SCENARIOS

Test Case Id	MDO_PT_TEST_001		Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
To verify that the users can view a video.	1	The user opens the MOBILE.OLD application and selects the Physical Training service.	The Physical Training service main screen is displayed.
	2	The user selects the first item of the „Video Categories“ list.	The video list of the selected category is displayed.
	3	The user highlights the first item from the video list.	The selected video's thumbnail picture and metadata are displayed.
	4	The user selects the first item from the video list.	The video screen is displayed and the video starts playing.

Test Case Id		MDO_PT_TEST_001	Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
	5	The user presses the pause, stop and play buttons.	The video pauses, stops and resumes playback respectively.
	6	The user presses the back button.	The service's main screen is displayed.

Test Case Id		MDO_PT_TEST_002	Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
To verify that the users can schedule a video.	1	The user opens the MOBILE.OLD application and selects the Physical Training service.	The Physical Training service main screen is displayed.
	2	The user selects the first item of the „Video Categories“ list.	The video list of the selected category is displayed.
	3	The user highlights the first item from the video list.	The selected video's thumbnail picture and metadata are displayed.
	4	The user presses the menu button and selects the "Schedule video" option.	A dialog is displayed prompting the user to select a date and a time.
	5	The user selects a date from the dialog and presses the "Set" button.	The dialog is hidden and a message box informs the user that the video has been scheduled.

Test Case Id		MDO_PT_TEST_003	Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
To verify that the users can view and edit their schedule.	1	The user opens the MOBILE.OLD application and selects the Physical Training service.	The Physical Training service main screen is displayed.
	2	The user presses the menu button and selects the "My Schedule" option.	The My Schedule screen is shown.

Test Case Id		MDO_PT_TEST_003	Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
	3	The user navigates to a scheduled video and presses the menu key and selects the "Remove video" option.	The video is removed from the user's schedule.

Test Case Id		MDO_PT_TEST_004	Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
To verify that the users can set a reminder for a scheduled video.	1	The user opens the MOBILE.OLD application and selects the Physical Training service.	The Physical Training service main screen is displayed.
	2	The user presses the menu button and selects the "My Schedule" option.	The My Schedule screen is shown.
	3	The user navigates to a scheduled video and presses the menu key and selects the "Add reminder" option.	A message box informs the user that a reminder has been set for the selected video.
	4	The user waits to be reminded via his mobile endpoint.	The mobile endpoint makes a ringing sound and displays a message informing the user that he has a scheduled video.

Test Case Id		MDO_PT_TEST_005	Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
To verify that the users can view videos relevant to their physical status.	1	The user opens the MOBILE.OLD application and selects the Physical Training service.	The Physical Training service main screen is displayed.
	2	The user presses the menu button and selects the "Settings" option.	The Settings screen is shown.
	3	The user selects his physical status from the screen and presses the back button.	The main screen is displayed.

Test Case Id		MDO_PT_TEST_005	Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
	4	The user navigates to the available videos.	The displayed videos are for the user's or for worse physical statuses.

## 10.4 TRANSNATIONAL JOURNEY PLANNING SERVICE

### 10.4.1 TESTS CONFIGURATION

The Transnational Journey Planning service will be tested using an IPO-85X IP Android STB for the TV endpoint. For the Smart Phone endpoint of the mobile environment, the service will be tested using a Samsung Galaxy S Smart Phone device.

### 10.4.2 VALIDATION SCENARIOS

Test Case Id		MDO_TJP_TEST_001	Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
To verify that the users can plan a regional journey.	1	The user opens the MOBILE.OLD application and selects the Transnational Journey Planning service.	The Transnational Journey Planning service main screen is displayed.
	2	The user selects the „Regional“ item from the Journey type list.	The Regional Journey information panel prompts the user to type his destination address and to edit his origin address or his transportation means.
	3	The user types an address to the respective field and presses the “Calculate Journey” button.	A list of transit directions along with alternative routes is displayed.

Test Case Id		MDO_TJP_TEST_002	Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
To verify that the users can plan a national journey.	1	The user opens the MOBILE.OLD application and selects the Transnational Journey Planning service.	The Transnational Journey Planning service main screen is displayed.

Test Case Id		MDO_TJP_TEST_002	Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
	2	The user selects the „National“ item from the Journey type list.	The National Journey information panel prompts the user to type his destination city and address and to edit his origin address or his transportation means.
	3	The user types his destination city and address to the respective fields and presses the “Calculate Journey” button.	A list of transit directions along with alternative routes is displayed.

Test Case Id		MDO_TJP_TEST_003	Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
To verify that the users can plan a transnational journey.	1	The user opens the MOBILE.OLD application and selects the Transnational Journey Planning service.	The Transnational Journey Planning service main screen is displayed.
	2	The user selects the „Transnational“ item from the Journey type list.	The Transnational Journey information panel prompts the user to type his destination country, city and address and to edit his origin address or his transportation means.
	3	The user types his destination country, city and address to the respective fields and presses the “Calculate Journey” button.	The Transnational Journey information panel displays an “Origin to airport” button and an “Airport to destination” button.
	4	The user presses the “Origin to airport” button from the Transnational Journey information panel.	A list of transit directions along with alternative routes for the nearest to the user’s origin airport is displayed.
	5	The user presses back and presses the “Airport to destination” button from the Transnational Journey information panel.	A list of transit directions along with alternative routes from the nearest airport to the user’s destination.

Test Case Id	MDO_TJP_TEST_004		Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
To verify that the users can configure their origin address and preferred transportation means.	1	The user opens the MOBILE.OLD application and selects the Transnational Journey Planning service.	The Transnational Journey Planning service main screen is displayed.
	2	The user presses the „Configuration“ button and is transferred to the configuration screen.	The configuration screen is shown and displays the user’s current origin address and preferred transportation means.
	3	The user changes his origin address and preferred transportation means, and presses the back button.	The origin and transportation means field in the regional, national and transnational journey information panels contain the same values that were added in the configuration screen.

## 10.5 ORIENTATION AND MOBILITY TRAINING SERVICE

### 10.5.1 HIKING APPLICATION

#### 10.5.1.1 TESTS CONFIGURATION

The Hiking Application will be tested using an Android STB for the TV endpoint. For the Smart Phone endpoint of the mobile environment, the service will be tested using a Samsung Galaxy S Smart Phone device.

#### 10.5.1.2 VALIDATION SCENARIOS

Test Case Id	MDO_HK_TEST_001		Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
To verify that the users can record a track using the mobile endpoint.	1	The user opens the MOBILE.OLD application and selects the Hiking Application.	The Hiking Application main screen is displayed.
	2	The user starts a new track by pressing the start button.	The button becomes a stop button.

Test Case Id		MDO_HK_TEST_001		Comments
Test Purpose	Test Case Description		Test Case Pass Criteria	
	3	The user starts walking with his mobile device.	The speed and direction of the user are displayed. A line corresponding to the path that the user is following is drawn on the screen.	
	4	The user presses the stop button.	A popup prompting the user to type a name for the track is displayed. After that, the stop button becomes a start button again.	

Test Case Id		MDO_HK_TEST_002		Comments
Test Purpose	Test Case Description		Test Case Pass Criteria	
To verify that the users can view a track using the mobile endpoint.	1	The user opens the MOBILE.OLD application and selects the Hiking Application.	The Hiking Application main screen is displayed.	
	2	The user selects a pre-recorded track from the „My tracks“ list.	A map is displayed showing the selected track.	
	3	The user selects the „Statistics“ button from the application’s menu bar.	The statistics screen is shown, displaying the track’s average and maximum speed, average and maximum elevation, total distance and time.	

Test Case Id		MDO_HK_TEST_003		Comments
Test Purpose	Test Case Description		Test Case Pass Criteria	
To verify that the users can view a track using the TV endpoint.	1	The user uses the mobile endpoint to record a track using the Hiking Application.	N/A	
	2	The user opens the MOBILE.OLD application in the TV endpoint and selects the Hiking Application.	The Hiking Application main screen is displayed. The screen’s tracks list contains the track that was recorded in the previous step.	

Test Case Id	MDO_HK_TEST_003		Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
	3	The user selects a track from the main screen's track list.	A map is displayed showing the selected track and its statistics. The displayed statistics include the track's average and maximum speed, average and maximum elevation, total distance and time.

## 10.5.2 ORIENTATION AND GEOTAG APPLICATION

### 10.5.2.1 TESTS CONFIGURATION

This service is tested using a Samsung Galaxy Note Smart Phone, on which the orientation and geotag application is installed. The tests must be conducted in an area with GPS coverage (outdoors), where a sufficient number of POIs is available in the surrounding area (e.g. a city of moderate size).

### 10.5.2.2 VALIDATION SCENARIOS

Test Case Id	MDO_OGA_TEST_001		Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
To verify that the user can take geotagged photos on their Smart Phone, and that geotagged photos are displayed correctly on a map.	1	The user opens the orientation and geotag app.	The app launches properly.
	2	The user selects the menu option for taking a photo.	The menu option for taking a photo is available.
	3	The user taps the take picture button as soon as the desired subject of the picture is visible in the viewfinder.	An on-screen viewfinder displays the scenery in front of the Smart Phone's main camera (not front camera).
	4	The user observes the new photo.	The photo looks like what was visible in the viewfinder at the time of taking it.
	5	The user navigates to the photo map by selecting the view photos option in the main menu.	On the photo map, a thumbnail of the newly taken photo at the current location of the user indicates that the photo has been geotagged correctly.

Test Case Id		MDO_OGA_TEST_001	Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
	6	The user taps the thumbnail of the newly taken photo.	The photo is enlarged and looks like it did shortly after it was taken.

Test Case Id		MDO_OGA_TEST_002	Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
Verifies that searching points of interest (POI) by category on a Smart Phone yields the results in the surrounding area with a variable radius.	1	The user selects the search nearby option in the main menu.	A list of frequently used POI categories is displayed, along with a text field, allowing the definition of a custom search term.
	2	The user selects one of the predefined POI categories.	All applicable POIs within a certain radius are displayed in a result list. Tapping a result item reveals additional information.
	3	The user taps the map tag.	All results from the list are displayed as markers on the map. If a result item was previously selected in the result list, it is now highlighted on the map.
	4	The user zooms out of the map.	Zooming out increases the search radius, so that applicable POIs from a larger surrounding area are included in the results.

Test Case Id		MDO_OGA_TEST_003	Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
Verifies that the user can navigate to a POI selected from the results of a custom search on a Smart Phone.	1	The user selects the search nearby option in the main menu.	A list of frequently used POI categories is displayed, along with a text field, allowing the definition of a custom search term.

Test Case Id		MDO_OGA_TEST_003	Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
	2	The user enters a search term into the provided text field and submits the input.	All POIs relevant to the search term within a certain radius are displayed in a result list. Tapping a result item reveals additional information and allows navigation.
	3	The user taps a search result, either in the result list or the alternative result map, and taps the navigate button.	The map is displayed, showing the route from the current position to the desired destination, including waypoints.
	4	The user switches from the map to the waypoint list by tapping the list tab.	The list is equivalent to the map, as it shows the directions in sequence.
	5	The user follows the directions until reaching the destination.	Route and directions adapt in accordance with the user's current position.
	6	Upon reaching the destination, the user selects the recently visited locations option from the main menu.	The destination of the just completed navigation is listed among earlier destinations.

Test Case Id		MDO_OGA_TEST_004	Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
Verifies that the user can save POIs for later navigation on a Smart Phone.	1	Upon performing a POI search, the user taps a search result, either in the result list or the alternative result map, and taps the memorize button.	Upon memorizing, the memorize button is replaced by an un-memorize button.
	2	The user selects the memorized locations option from the main menu.	The previously POI is displayed among other memorized locations in a list, or an equivalent map. The memorized location can be deleted or used for navigation.

Test Case Id		MDO_OGA_TEST_005	Comments
Test Purpose	Test Case Description		Test Case Pass Criteria

Test Case Id		MDO_OGA_TEST_005	Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
Verifies that the user can view all POIs in their surrounding area on a Smart Phone.	1	The user selects the surrounding area option from the main menu.	A map is displayed with POIs of all categories within a certain radius, which is governed by the map's zoom level.
	2	The user switches to the result list by tapping the list tab.	The surrounding POIs are displayed in a list form equivalent to the map. Tapping a result item allows navigating and memorizing the POI.

### 10.5.3 INDOOR / OUTDOOR ORIENTATION TRAINING APPLICATION

#### 10.5.3.1 TESTS CONFIGURATION

The Indoor / outdoor orientation service will be tested using an Android STB for the TV endpoint. For the Smart Phone endpoint of the mobile environment, the service will be tested using a Samsung Galaxy S Smart Phone device.

#### 10.5.3.2 VALIDATION SCENARIOS

Test Case Id		MDO_IOO_TEST_001	Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
To verify that the users can view indoor locations.	1	The user opens the MOBILE.OLD application and selects the Indoor/ outdoor orientation service.	The Indoor / outdoor orientation service main screen is displayed.
	2	The user selects the „Museums“ item from the points of interest categories list.	The contents of the „Museum“ category are displayed in the „Museum POIs“ panel.
	3	The user selects the “Palace of Versailles” item from the Museum POIs panel.	The Google Street View application is displayed showing the inside of the palace of Versailles.

Test Case Id		MDO_IOO_TEST_002	Comments
Test Purpose	Test Case Description		Test Case Pass Criteria

Test Case Id		MDO_IOO_TEST_002	Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
To verify that the users can view outdoor locations.	1	The user opens the MOBILE.OLD application and selects the Indoor/ outdoor orientation service.	The Indoor / outdoor orientation service main screen is displayed.
	2	The user selects the „Landmarks“ item from the points of interest categories list.	The contents of the „Landmarks“ category are displayed in the „Landmark POIs“ panel.
	3	The user selects the “La Sagrada Familia” item from the Museum POIs panel.	The Google Street View application is displayed and shows La Sagrada Familia.

Test Case Id		MDO_IOO_TEST_003	Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
To verify that the users can configure the displayed points of interest categories.	1	The user opens the MOBILE.OLD application and selects the Indoor/ outdoor orientation service.	The Indoor / outdoor orientation service main screen is displayed.
	2	The user presses the configuration button and is transferred to the configuration screen.	The configuration screen is displayed.
	3	The user checks the POI categories that he wants to be displayed and presses the back button.	The service’s main screen is displayed and the point of interest categories list contains only the POI categories that were checked by the user.

## 10.5.4 FIRST AID APPLICATION

### 10.5.4.1 TESTS CONFIGURATION

The mobile application is tested on a Samsung Galaxy Note Smart Phone. In order to simulate dispatch of emergency calls, a debug version of the mobile application should be used, where no actual emergency number is dialled in order to avoid legal issues.

### 10.5.4.2 VALIDATION SCENARIOS

Test Case Id		MDO_FAA_TEST_001	Comments
Test Purpose	Test Case Description		Test Case Pass Criteria

Test Case Id		MDO_FAA_TEST_001	Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
Verifies that the user can dispatch emergency calls on a Smart Phone.	1	The user opens the first aid application	The app launches successfully; a list of first aid topics is visible, as well as a highlighted emergency button.
	2	The user selects the emergency button.	A list of emergency services including their telephone numbers is shown, which depends on the current location of the user.
	3	The user taps the emergency service appropriate for the situation.	A dialog is shown, requesting confirmation for calling the emergency number.
	4	The user confirms or denies calling the emergency number.	If confirmed, a call to the respective emergency service is initiated. Otherwise, the user returns to the emergency number overview.

Test Case Id		MDO_FAA_TEST_002	Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
Verifies that the user can read guides for emergency situations on a Smart Phone.	1	The user opens the first aid application	The app launches successfully; a list of first aid topics is visible, as well as a highlighted emergency button.
	2	The user selects a first aid topic of interest.	The first page of the first aid guide is shown.
	3	The user taps the read-aloud button.	The content of the guide page is read out loud using a speech synthesizer.
	4	The user taps the next button.	The guide switches to the next page in sequence.
	5	If available, the user taps the additional information button.	Additional information concerning the current step in the guide is displayed.

Test Case Id	MDO_FAA_TEST_002		Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
	6	The user taps the next button until the last page of the guide has been reached.	The next button is replaced by a return to menu button.
	7	The user taps the return to menu button.	The first aid app's main menu is displayed.

## 10.5.5 CHECKLIST APPLICATION

### 10.5.5.1 TESTS CONFIGURATION

This application is tested on a Samsung Galaxy Note Smart Phone.

### 10.5.5.2 VALIDATION SCENARIOS

Test Case Id	MDO_CLA_TEST_001		Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
To verify that new checklists can be created on a Smart Phone based on topic-specific templates.	1	The user opens the checklist application.	The checklist app opens successfully and displays a list of existing checklists, if available.
	2	The user taps the create checklist button.	The checklist creation view is displayed, providing a list of templates, as well as the possibility to create a custom checklist via text input and icon selection.
	3	The user taps the checklist template of their choice.	Icon and title of the template list are now displayed in the custom checklist input elements for modification.
	4	The user taps the create checklist button.	The checklist is shown. Depending on the template, it may be populated by some pre-defined items.

Test Case Id	MDO_CLA_TEST_002		Comments
Test Purpose	Test Case Description		Test Case Pass Criteria

Test Case Id		MDO_CLA_TEST_002	Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
To verify that new predefined items can be added to an existing checklist on a Smart Phone.	1	The user taps an existing checklist in the main menu, which is based on a template.	The checklist and its items are displayed.
	2	The user taps the add item button.	The item creation view is shown, offering possibilities to enter text, take a photo, record audio, or choose a predefined item according to the selected template.
	3	The user selects one of the predefined items.	Icon and title of the predefined item are now displayed in the custom item input elements for modification.
	4	The user taps the add to checklist button.	The checklist is displayed, which now includes the newly created item.

Test Case Id		MDO_CLA_TEST_003	Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
To verify that new text-based items can be added to an existing checklist on a Smart Phone.	1	The user taps an existing checklist in the main menu, which is based on a template.	The checklist and its items are displayed.
	2	The user taps the add item button.	The item creation view is shown, offering possibilities to enter text, take a photo, record audio, or choose a predefined item according to the selected template.
	3	The user enters text into the text input field.	The item text is updated.
	4	The user taps the icon field.	A collection of predefined icons is displayed.

Test Case Id		MDO_CLA_TEST_003	Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
	5	The user selects one of the icons.	The item creation view is displayed again, now with the selected icon in the icon field.
	4	The user taps the add to checklist button.	The checklist is displayed, which now includes the newly created item with the selected text and icon.

Test Case Id		MDO_CLA_TEST_004	Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
To verify that new photo items can be added to an existing checklist on a Smart Phone.	1	The user taps an existing checklist in the main menu, which is based on a template.	The checklist and its items are displayed.
	2	The user taps the add item button.	The item creation view is shown, offering possibilities to enter text, take a photo, record audio, or choose a predefined item according to the selected template.
	3	The user selects the option to create a new photo item.	A viewfinder is displayed, showing what the main camera (not front camera) is currently pointed at.
	4	The user taps the take photo button.	A confirmation dialogue, containing the newly taken photo, is shown, whether the photo should be accepted or retaken redone (in which case this step is repeated).
	5	The user accepts the photo by tapping the appropriate button.	The item creation view is shown again. The newly taken photo is now visible in the icon input element.

Test Case Id	MDO_CLA_TEST_004		Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
	6	The user taps the add to checklist button.	The checklist is displayed, which now includes the newly created item, including the newly taken photo as its icon.

Test Case Id	MDO_CLA_TEST_005		Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
To verify that new audio items can be added to an existing checklist on a Smart Phone.	1	The user taps an existing checklist in the main menu, which is based on a template.	The checklist and its items are displayed.
	2	The user taps the add item button.	The item creation view is shown, offering possibilities to enter text, take a photo, record audio, or choose a predefined item according to the selected template.
	3	The user selects the option to create a new audio item.	An audio recorder is displayed, containing a button to explicitly start the recording.
	4	The user presses the record button, speaks a message, and then presses the stop recording button.	After recording, a dialog is provided for listening to the audio. The recording can be accepted, or redone (in which case this step is repeated).
	5	The user accepts the recording by tapping the appropriate button.	The item creation view is shown again.
	6	The user taps the add to checklist button.	The checklist is displayed, which now includes the newly created item. Its icon signifies that this is an audio recording.
	7	The user taps the icon of the newly created item.	The previously recorded audio is replayed.

Test Case Id		MDO_CLA_TEST_006		Comments
Test Purpose	Test Case Description		Test Case Pass Criteria	
To verify that checklist items can be checked and unchecked on a Smart Phone.	1	The user taps an existing checklist in the main menu, which has more than one item.	The checklist and its items are displayed.	
	2	The user taps the checkbox next to the item to check it.	The item's text and icon are crossed out, the checkbox is checked, the item's colours are dimmed, and it is moved to the bottom of the list.	
	3	The user taps the checkbox next to the previously checked item to uncheck it.	The box is unchecked, the colours and layout of the item are restored, and the item is moved back above the checked items at the end of the list.	

## 10.5.6 ORIENTATION AND MOBILITY GAMES

### 10.5.6.1 TESTS CONFIGURATION

This service is tested using a Samsung Galaxy Note Smart Phone, on which the scavenger hunt application is installed. The tests must be conducted in an area with GPS coverage (outdoors), where a sufficient number of POIs is available in the surrounding area (e.g. a city of moderate size). A test configuration of scavenger hunts is used for testing.

### 10.5.6.2 VALIDATION SCENARIOS

Test Case Id		MDO_OMG_TEST_001		Comments
Test Purpose	Test Case Description		Test Case Pass Criteria	
Verifies that users can start a scavenger hunt using their Smart Phone, where automatically POIs are combined for a route and personal POIs can be	1	The user opens the scavenger hunt application.	The app launches successfully. A main menu is displayed, offering options to start a new game and view previous scores, to define new personal POIs and to open the settings view.	

Test Case Id		MDO_OMG_TEST_001	Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
integrated	2	The user selects the menu option to start a new game.	A configuration view is shown where the user can optionally specify an approximate available duration and personal necessary POIs.
	3	Via "Start button" the user can start the game.	The game starts.

Test Case Id		MDO_OMG_TEST_002	Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
Verifies that a scavenger hunt can be completed on a Smart Phone.	1	The user configures and starts a scavenger hunt.	Direction and distance to the next destination, as well as distance to the final destination and the estimated total remaining time are displayed. A help button offers navigation on demand.
	2	The user walks part of the scavenger hunt.	The values are updated periodically in accordance with the user's current position.
	3	The user presses the help button.	A map with a route to the next destination is displayed. The user's current position is indicated. An alternative list representation with written walking directions is available as well.
	4	The user reaches a destination.	A quiz question related to the destination is displayed, offering a number of possible answers.
	5	The user selects the most plausible answer.	The correct answer is highlighted. A button is provided to continue with the scavenger hunt.
	6	The user taps the button to continue the scavenger hunt.	Analogous behaviour to steps 1 through 5.

Test Case Id		MDO_OMG_TEST_002	Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
	7	The user taps the button to finish the scavenger hunt after answering the final destination's quiz question.	The score, actual walking time, and actual walking distance are displayed.

Test Case Id		MDO_OMG_TEST_003	Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
Verifies that a user can view previous played games.	1	The user opens the scavenger hunt application.	The app launches successfully. A main menu is displayed, offering options to start a new game and view previous scores, to define new personal POIs and to open the settings view.
	2	The user selects the menu option to show previous games	A list of previous games is shown
	3	The user selects a certain game	Information according the selected game is shown.
	4	The user can select different information tabs (list, map, questions)	The selected game information is shown.

Test Case Id		MDO_OMG_TEST_004	Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
Verifies that a user can adapt the personal POI's.	1	The user opens the scavenger hunt application.	The app launches successfully. A main menu is displayed, offering options to start a new game and view previous scores, to define new personal POIs and to open the settings view.
	2	The user selects the menu option to define new personal POIs.	The list of actual defined personal POIs is shown
	3	The user can select a defined POI.	Information according the selected POIs is shown.

Test Case Id		MDO_OMG_TEST_004	Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
	4	The user can adapt or delete the selected POI	The list of actual defined personal POIs is shown

Test Case Id		MDO_OMG_TEST_005	Comments
Test Purpose	Test Case Description		Test Case Pass Criteria
Verifies that a user can add a new personal POI.	1	The user opens the scavenger hunt application.	The app launches successfully. A main menu is displayed, offering options to start a new game and view previous scores, to define new personal POIs and to open the settings view.
	2	The user selects the menu option to define new personal POIs.	The list of actual defined personal POI's is shown
	3	The user can select the button for a new POI.	The default information according the new POI is shown.
	4	The user can call the map to find the new POI and select it	The information according the selected POI is shown.
	5	The user saves the new POI.	The new personal POI is stored.

## 11. CONCLUSIONS

The MOBILE.OLD services that will be developed in the context of the project have been specified. All the requirements that were presented in the D2.1 deliverable have been considered and allocated to the various subsystems of each service. In summary, the MOBILE.OLD services include:

- Personalized Transportation News Headlines Service, which will display regional, national and European transportation news content to the user, in the areas of ground and air transportation, traffic and strikes information and weather.
- Geofencing Service, which will provide mobility assistance to persons suffering from cognitive impairments while performing outdoor activities by using positioning monitoring and automatic alert dispatching.
- Physical Training Service, which will provide on-demand multimedia exercises to the users, allowing them to enhance their physical status.
- Transnational Journey Planning Service, which will allow the user to select an origin and a destination location, along with a transportation means, and will provide travel directions between them.
- Orientation and Mobility Training Service, which will allow the user to improve his orientation by providing a set of applications to enhance various outdoor activities, such as the Hiking Application, the Geotag Application and the Orientation and Mobility Games application. Also, the service will provide applications that will help the user better prepare for a travel, with the Indoor / outdoor Orientation Training and the Checklist applications or to even prepare for emergencies with the First Aid application.

Also, this document provides the architecture of the MOBILE.OLD Application Server that will allow the various services to provide a consistent user experience by sharing user data and preferences.

Finally, this deliverable describes the evaluation and deployment plan of the MOBILE.OLD services which includes the user selection and training, system installation and prototype evaluations, along with the specific validation plan of each service.

## 12. REQUIREMENTS COVERAGE MATRIX

The following table provides a requirements traceability matrix.

Requirement ID (from D 2.1)	Requirement addressed in D 2.2	Comments
MOBILE.OLD-A-1	Yes, section 2	
MOBILE.OLD-A-2	Yes, section 2	
MOBILE.OLD-A-3	Yes, section 2	
MOBILE.OLD-A-4	Yes, section 2	
MOBILE.OLD-A-5	Yes, section 5 and 9	
MOBILE.OLD-A-6	Yes, section 7	
MOBILE.OLD-A-7	Yes, section 7	
MOBILE.OLD-A-8	Yes, section 2	
MOBILE.OLD-A-9	Yes, section 4	
MOBILE.OLD-C-1	No	This requirement is not explicitly mentioned in D2.2, but will be fulfilled anyway by the Smart Phone and its internal memory.
MOBILE.OLD-C-2	Yes, section 2 and section 7.1	
MOBILE.OLD-C-3	Yes, section 7.1	
MOBILE.OLD-C-4	Yes, section 7.1	
MOBILE.OLD-C-5	Yes, section 4	
MOBILE.OLD-I-1	Yes, section 4.1	
MOBILE.OLD-I-2	Yes, section 4.1	
MOBILE.OLD-I-3	Yes, section 4.1	
MOBILE.OLD-I-4	Yes, section 4.1	
MOBILE.OLD-I-5	Yes, section 4.4	
MOBILE.OLD-I-6	No	The services will be supported in the pilot

		sites languages.
<b>MOBILE.OLD-I-7</b>	Yes, section 4.2	
<b>MOBILE.OLD-I-8</b>	Yes, section 4.2	
<b>MOBILE.OLD-I-9</b>	No	Not addressed so far, it will be addressed as much as possible in D.4.2.
<b>MOBILE.OLD-I-10</b>	Yes, partially section 4.3	It was decided among the consortium that the exercises will not be proposed based on the selected trip.
<b>MOBILE.OLD-I-11</b>	Yes, section 4.3	
<b>MOBILE.OLD-I-12</b>	No	It was decided based on the service ratings of the end-users that this service will not be implemented in the context of this project.
<b>MOBILE.OLD-I-13</b>	Yes, section 4.5.5.	
<b>MOBILE.OLD-I-14</b>	Yes, section 4.5.2.	
<b>MOBILE.OLD-I-15</b>	Yes, section 4.5.2	
<b>MOBILE.OLD-I-16</b>	Yes, section 4.5.1	
<b>MOBILE.OLD-I-17</b>	Yes, partially section 4.5.4	This service will only be available on the Smart Phone.
<b>MOBILE.OLD-I-18</b>	No	It was decided based on the service ratings of the end-users that this service will not be implemented in the context of this project.
<b>MOBILE.OLD-S-1</b>	Yes, section 7.1	
<b>MOBILE.OLD-S-2</b>	Yes, section 7.1	
<b>MOBILE.OLD-S-3</b>	Yes, section 7.1	
<b>MOBILE.OLD-S-4</b>	Yes, section 2	
<b>MOBILE.OLD-S-5</b>	Yes, section 7.1	
<b>MOBILE.OLD-T-1</b>	No	Out of scope of MOBILE.OLD.
<b>MOBILE.OLD-T-2</b>	Yes, section 7	
<b>MOBILE.OLD-T-3</b>	Yes, section 7	

<b>MOBILE.OLD-AUT-I-3</b>	Yes, partially. Section 6.1.	More details will be provided in D.4.2
<b>MOBILE.OLD-AUT-I-2</b>	Not addressed so far.	More details will be provided in D.4.2.
<b>MOBILE.OLD-AUT-RO-SPA-I-3</b>	Yes, section 4	
<b>MOBILE.OLD-AUT-SPA-I-4</b>	Yes, section 9.2.4	
<b>MOBILE.OLD-AUT-SPA-I-5</b>	Not addressed so far.	Not addressed so far, it will be addressed in D.4.2.
<b>MOBILE.OLD-AUT-NL-RO-I-6</b>	Not addressed so far.	Not addressed so far, it will be addressed in D.4.2.
<b>MOBILE.OLD-NL-I-7</b>	Not addressed so far.	Not addressed so far, it will be addressed in D.4.2.
<b>MOBILE.OLD-NL-I-8</b>	Not addressed so far.	Not addressed so far, it will be addressed in D.4.2.
<b>MOBILE.OLD-NL-I-9</b>	Not addressed so far.	Not addressed so far, it will be addressed in D.4.2.
<b>MOBILE.OLD-AUT-SPA-I-10</b>	Yes, section 6.1	
<b>MOBILE.OLD-NL-I-11</b>	Yes, section 5.2	
<b>MOBILE.OLD-NL-I-12</b>	Not addressed so far.	Not addressed so far, it will be addressed in D.4.2.
<b>MOBILE.OLD-RO-I-13</b>	Not addressed so far.	Not addressed so far, it will be addressed in D.4.2.
<b>MOBILE.OLD-RO-I-14</b>	Not addressed so far.	Not addressed so far, it will be addressed in D.4.2.
<b>MOBILE.OLD-RO-I-15</b>	Yes, section 4	
<b>MOBILE.OLD-RO-I-16</b>	Not addressed so far.	Not addressed so far, it will be addressed in D.4.2.
<b>MOBILE.OLD-AUT-RO-S-1</b>	No	It was decided based on the service ratings of the end-users that this service will not be implemented in the context of this project.
<b>MOBILE.OLD-AUT-NL-RO-S-2</b>	No	Hard to assess at this stage.
<b>MOBILE.OLD-AUT-S-3</b>	No	This requirement is not explicitly mentioned in D2.2, but will be fulfilled

		anyway since the services will be web-based.
<b>MOBILE.OLD-AUT-SPA-S-4</b>	No	See section 6.1
<b>MOBILE.OLD-NL-S-5</b>	No	No interface with social networks was decided.
<b>MOBILE.OLD-RO-S-6</b>	Yes, section 4.3	
<b>MOBILE.OLD-AUT-NL-RO-SPA-S-7</b>	Yes, section 9.2.6	
<b>MOBILE.OLD-NL-S-8</b>	Not addressed so far.	Should be addressed during the business model definition.
<b>MOBILE.OLD-NL-S-9</b>	Yes, section 4.4	
<b>MOBILE.OLD-RO-S-10</b>	Yes, section 4.4	
<b>MOBILE.OLD-NL-S-11</b>	Yes, section 4.4	
<b>MOBILE.OLD-NL-RO-S-12</b>	No	It was decided based on the service ratings of the end-users that this service will not be implemented in the context of this project.
<b>MOBILE.OLD-RO-S-13</b>	No	It was decided among the consortium that the function for detecting a possible user fall will only be implemented as a demo.
<b>MOBILE.OLD-RO-S-14</b>	Yes, 4.5.5	
<b>MOBILE.OLD-RO-S-15</b>	Yes, 4.5.2	
<b>MOBILE.OLD-NL-S-16</b>	Yes, section 4.5.4	
<b>MOBILE.OLD-RO-S-17</b>	Not addressed so far.	Should be addressed during the business model definition.
<b>MOBILE.OLD-AUT-NL-SPA-C-1</b>	Yes, section 7.1	
<b>MOBILE.OLD-RO-C-2</b>	No	Elderly individuals with heart pacemakers are admonished to not carry the Smart Phone in shirt pockets.
<b>MOBILE.OLD-RO-C-3</b>	No	Out of scope of MOBILE.OLD
<b>MOBILE.OLD-SP-C-4</b>	No	The Smart Phone locking function should be used.

**Table 3: Requirements traceability matrix**

## 13. ANNEX A: WEB SERVICE TECHNOLOGY EVALUATION

### 13.1 INTRODUCTION

The MOBILE.OLD services require some sort of off-client-device storage facility to persist data, transmit it between devices, and broadcast content. RESTful web services are especially suitable for highly data-oriented operations. In comparison to action-oriented SOAP services, they are much more lightweight by using URLs to identify resources, and HTTP verbs to determine the operation – usually POST, GET, PUT, DELETE for create, read(-all), update and delete (CRUD), respectively.

This document discusses several technology alternatives, which are evaluated using equivalent prototypes. Finally in section 13.4, an informed recommendation is made based on the information acquired in the course of the evaluation.

Although server side development will be initially performed independently by the technical partners for the most part, using similar technological approaches reduces the effort required by the integration phase later in the project, under the condition that integration on a higher level than database level is desirable.

Noteworthy implementation details and caveats are described in section 13.2. Section 13.3 analyses the approaches in terms of performance, resource efficiency and difficulty of development.

All technologies discussed in this document are, unless specified otherwise, industry-proven, open source and available for commercial use at no charge.

It is assumed that the reader is acquainted with web service methodologies, network protocols and basic design patterns at least on a conceptual level.

#### 13.1.1 METHODOLOGY

To compare the selected technology alternatives, each was used to implement a basic RESTful web service for user management, supporting create, read, update and delete operations. Authentication and authorization were omitted and will be discussed in theory.

All implementations use PostgreSQL<sup>2</sup> as relational database management system (RDBMS), which is a leading open source database capable of storing and accessing large amounts of data at high speeds. The choice was made to even out performance differences that would exist between the various solutions if different RDBMS were used. All presented technology alternatives are actually abstract enough that the backend database can be exchanged without modification to the codebase by adapting a configuration file and supplying an appropriate database driver package. This allows using virtually any popular RDBMS.

A performance test is conducted with each of the implementations to determine the throughput of each approach in terms of requests per second.

---

<sup>2</sup> <http://www.postgresql.org/>

Implementation subtleties of each approach, as well as the overall effort necessary to become acquainted with the technology and its complexity are documented. Since high complexity increases the probability of issues, this factor must be taken into account – after all, additional time spent on fixing problems directly increases the cost of an implementation.

The final “score” of each technology alternative is determined by considering performance, resource efficiency, and difficulty of development.

### 13.1.2 REQUIREMENT OUTLINE

In order to determine the most suitable web service technology for the MOBILE.OLD project, the role of the web service in the network of MOBILE.OLD applications must be clarified.

The principal duties of the MOBILE.OLD web service are:

- Saving and serving user settings, including UI customizations
- Saving and serving user content
- Serving content created by the service provider
- Processing and serving content fetched from third party web services

Considering the nature of the MOBILE.OLD services (not to be confused with *web* services) that are being implemented, the expected frequency of requests is rather low compared to, for example, real time games. Read operations are more likely than write operations, which allows efficient caching.

However, whether caching would actually be necessary remains questionable. Given the target audience, the MOBILE.OLD services can't be expected to draw a number of users comparable to a hot web startup, especially in the prototype phases.

Ideally, the web service should support JSON as content type. Since it's more lightweight and easier to parse, using JSON would conserve processing power and battery life on mobile devices, which is an important consideration. JSON is also the preferred content type in JavaScript environments, and is easier to process for native Android apps in the absence of Java's powerful JAXB marshalling engine.

## 13.2 TECHNOLOGY STACKS

A section each is dedicated to describe the technology stacks that were investigated. The key facts are summarized at the conclusion. Of course, only a limited number of technologies could be investigated. Section 13.2.4 therefore argues the omission of candidates that would have at least satisfied the criteria of being widely used and industry-proven.

### 13.2.1 JAX-RS

JAX-RS was introduced in Java EE 6 and is a standardized Java API for RESTful web services. It's the sister standard of JAX-WS, which deals with classic SOAP services, and is based on JSR 311<sup>3</sup>.

#### 13.2.1.1 CHARACTERISTICS

---

<sup>3</sup> <http://jcp.org/en/jsr/detail?id=311>

JAX-RS is designed to be configured almost entirely through annotations applied to a plain old Java object (POJO), which serves as a resource. It is annotated with the resource base path. Paths specified for methods of the class are relative to this base path.

Methods, which are annotated with an HTTP verb (e.g. *@GET*) are automatically called when the resource is accessed using this HTTP method. If there are multiple methods handling the same HTTP method, they can be differentiated using annotations to specify the acceptable request content types, as well as the returned response types. Content type negotiation is performed automatically using the HTTP headers *Accept* and *ContentType*.

The request body can be unmarshalled automatically using JAXB, the standard Java serialization API. Likewise, it is possible to send any serializable object as a response by simply returning it – JAX-RS marshals it automatically.

URL-, header-, cookie-, form- and other parameters can be accessed by annotating a methods arguments.

JAX-RS is merely an API. Actual functionality conforming to the standard must be provided by a third party implementation. Jersey<sup>4</sup> is the reference implementation. Others include Apache CXF and Restlet.

#### 13.2.1.2 SECURITY

Annotations can be used to restrict request handling methods of a resource to specific roles. Alternatively, the security context can be injected by the JAX-RS container by using the *@Context* annotation on a *SecurityContext* instance variable or method argument. This allows manual access to the information about the authenticated user. The servlet configuration may also specify path-based access restrictions.

Authentication itself is performed by the servlet container, which is Tomcat in this case.

#### 13.2.1.3 PROTOTYPE

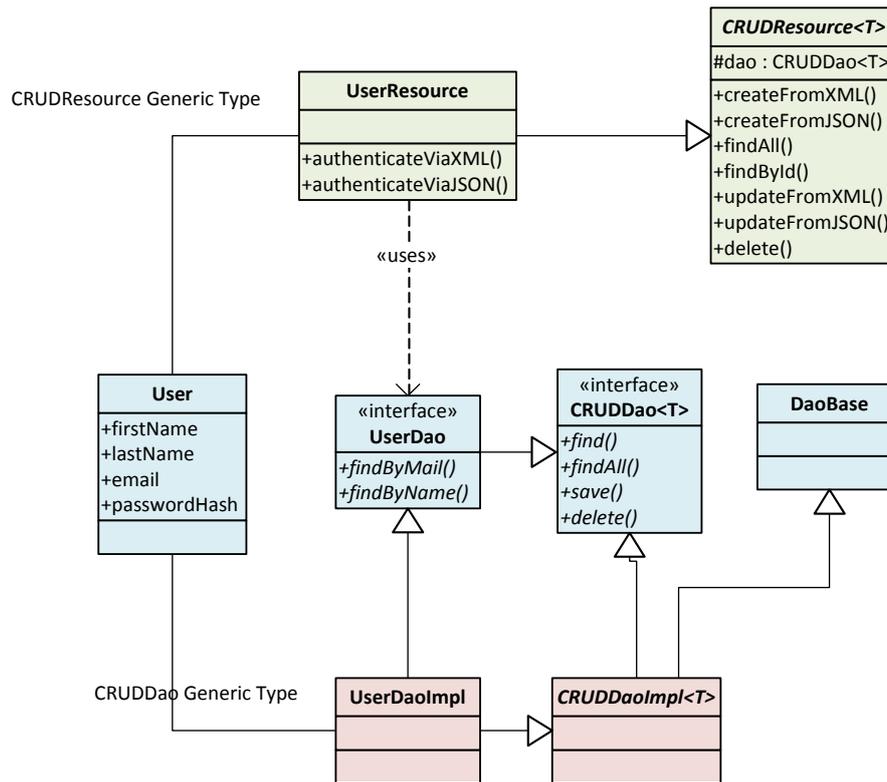
To evaluate JAX-RS, a user management web service is used, which runs on a Tomcat 7.0 application server. For accessing the database, the Java Persistence API (JPA). JPA is implemented by Hibernate, which uses the PostgreSQL JDBC driver to access the database. Jersey is used as JAX-RS implementation.

Figure 64 shows the architecture of the JAX-RS prototype used for this evaluation. In terms of the multi-layer architecture pattern, this prototype features the data tier, consisting of data access object (DAO) interfaces *UserDao* and *CRUDDao<T>*, and the domain class *User*, as well as database-specific implementations of the DAO interfaces *UserDaoImpl* and *CRUDDaoImpl<T>*.

The database-specific implementations are in fact JPA-specific, which reduces the effort required to switch between database management systems to changing a few lines in the persistence configuration file.

---

<sup>4</sup> <http://jersey.java.net/>



**Figure 64: Simplified architecture of the JAX-RS prototype**

*UserResource* and *CRUDResource* represent the logic tier. The presentation tier is not present in the architecture, as this task is delegated to the client accessing the web service.

Similarly to the DAO implementations, the resource classes use only features of the JAX-RS API – no distinct features of the JAX-RS implementation. This allows for easy exchange of JAX-RS implementations by adapting the servlet configuration.

The prototype support data transfer using XML and JSON. It should be noted that when trying to unmarshall JSON using JAX-RS' built-in mechanism, errors occur. Therefore, methods accepting data in the request body using either format have separate implementations, where the JSON implementation unmarshalls manually using Google's Gson library.

## 13.2.2 SPRING MVC

Spring MVC is a Java-based web application framework that is part of the greater Spring application framework. Originally, Spring consisted of an application container and a framework to build applications running on top of it. It evolved to support web applications following the MVC pattern, which are not hosted in a Spring application container, but a web application container adhering to the Java Servlet specification.

### 13.2.2.1 CHARACTERISTICS

The chief advantages of Spring MVC are inversion of control and aspect-oriented programming (AOP).

Inversion of control allows setting properties of Java beans through the application container by specifying in a configuration type, which classes should be instantiated and supplied to the bean. This

eliminates the overhead of setting properties manually. A subsequent advantage is that code is more loosely coupled.

For instance, a resource class might need one or more data access objects to access the database. Instead of manually creating instances whenever a new resource object is created, the resource container just needs a setter for a private reference to the DAO. The application container can be configured using XML to automatically create instances of the DAO and supplying it to the resource's setter. Because DAOs are mostly stateless, the container could even use a single instance of the DAO for all resource needing it (singleton pattern).

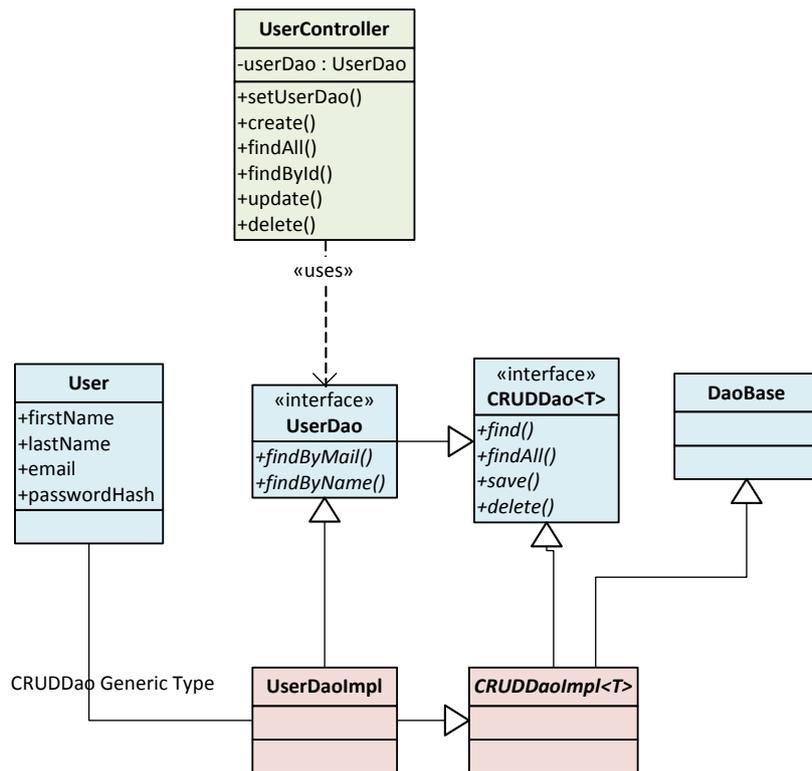
AOP allows executing code before or after the execution of a method, as well as in case of an exception while executing it. This programming paradigm is intended to separate cross-cutting concerns from the codebase. Cross-cutting concerns refer to tasks that require code concerning it to be scattered across the entire codebase. For example, a class to access the database needs to log messages, so the source code must contain calls to the logger. Changing the logging system would then require changing every single file using the logger. AOP would separate the logging logic from the database logic, so it can be maintained centrally. A configuration file would link a database access method to the logging method.

#### **13.2.2.2 SECURITY**

Spring contains a versatile security framework, which allows fine grained URL-based or manual authorization using XML configuration files. A resource's handling methods can access user information, including the role, in a manner similar to the JAX-RS approach.

#### **13.2.2.3 PROTOTYPE**

For a great part, the architecture of the JAX-RS approach, as visible in Figure 64, could be reused. The most significant change (see Figure 65) is that the resource class was replaced by a controller that does essentially the same, but integrates into Spring's MVC architecture. Note that for simplicity's sake, the MVC pattern is broken in this example, since the controller operates directly on the DAL instead of using the model.



**Figure 65: Simplified architecture of the Spring MVC prototype**

A notable improvement introduced by Spring is that the `userDao` property of `UserController` can be set with a `UserDao` instance created by the application container using XML configuration, a setter and Spring's inversion of control feature.

Again, Tomcat is used as application server.

### 13.2.3 DJANGO-PISTON

Django<sup>5</sup> is the leading Python web development framework. Django-Piston<sup>6</sup> is a RESTful web service library on top of it, which is developed internally at Bitbucket.

#### 13.2.3.1 CHARACTERISTICS

Django allows the development of MVC web applications. Object-relational mapping is supported out of the box, including automatic database schema generation. Backends for most popular DBMSes are available – this includes not only classic relational databases, but also NoSQL solutions.

The framework supports caching out of the box, including in-memory caching, memcached, database cache and custom cache implementations. There is also built-in support for client cache control via HTTP headers. Third party packages to interact with upstream caches, such as Varnish, are available as well.

Concerning content formats, Django-Piston allows easy (de-)serialization between python objects and the formats XML, JSON, YAML, Pickle (binary Python object serialization) and a XML-based format specialized for information exchange between Django instances. Custom formats can be

<sup>5</sup> <https://www.djangoproject.com/>

<sup>6</sup> <https://bitbucket.org/jespern/django-piston/wiki/Documentation>

added easily. The recommended method for content type negotiation is appending the format name like a file extension to the URL, like `/api/users.xml` or `/api/user/1.xml`.

Most REST-related work is performed by a class derived from Piston's *BaseHandler*. Overriding the *create*, *read*, *update* and *delete* methods assigns functionality to HTTP POST, GET, PUT and DELETE requests, respectively. The web application's routing configuration specifies which URL(s) are handled by which handler using regular expressions.

It is possible to restrict the number of calls to a certain handler methods or groups of them per user per timeframe using decorators (similar to Java annotations).

Translated to terms of the MVC pattern, the handler takes the role of the controller in the MVC pattern. The return values of the handling methods constitute the views. Classic Django model classes represent the model.

### 13.2.3.2 SECURITY

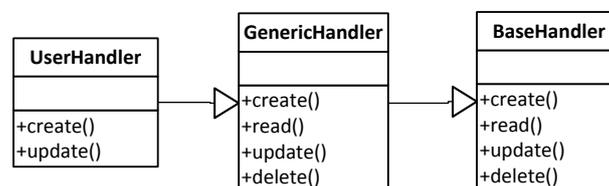
A great advantage of the Django-Piston approach is that user- and role management is available out of the box, including an administration web interface, an extensible database schema, and authentication/authorization facilities.

Authentication can be required by specifying an option in the routing configuration for the affected URL. As a result, all information about the authenticated user is available to the methods of the resource handler, allowing for fine grained privileges. The available authenticated modes on HTTP level are classic HTTP authentication and OAuth.

### 13.2.3.3 PROTOTYPE

A prototypical user management service is implemented using Django and Django-Piston, as well as `psycogp2`<sup>7</sup> as PostgreSQL database backend. For fair comparison with the other approaches, caching is disabled.

`uWSGI`<sup>8</sup> is used as web server. It is a high performance web application container adhering to the WSGI standard, which is a communication protocol bridging the gap between web servers and web applications. `uWSGI` is configured to use 4 worker processes to exploit the multicore processor on the test machine (see section 13.3.1.1 for details on which test hardware was used).



**Figure 66: Simplified architecture of the Django-Piston prototype**

Figure 66 shows the classes that constitute the main logic of the prototype. *BaseHandler* is provided by Django-Piston and contains empty implementations of the CRUD methods. *GenericHandler* implements those methods in a generic manner, which could be used for most typical Django

<sup>7</sup> <http://pypi.python.org/pypi/psycogp2>

<sup>8</sup> <https://github.com/unbit/uwsgi-docs>

models. For the actual *UserHandler*, *GenericHandler*'s method implementations are sufficient for all but the *create* and *update* methods, which need logic to hash the password.

### 13.2.4 DISREGARDED TECHNOLOGY STACKS

The commercial web application landscape is dominated by Enterprise Java on one side, and several scripting languages on the other. This document's focus on Java approaches to RESTful web services stems from the fact that the participating partners in the project are likely more familiar with those technologies, eliminating the need to learn something new from scratch. Django was still included as a representative of script language based web application development stacks in order to determine whether the effort of learning something new could pay off.

Below are major players in the area of web applications and –services, whose omission might be most notable:

- **Ruby on Rails**<sup>9</sup>: Web applications written using this popular Ruby-based framework are RESTful web services from the start. Much can be accomplished with little code. On the downside, this high level of abstraction introduces high complexity, a large number of dependencies, and performance penalties.
- **Node.js**<sup>10</sup>: The increasingly popular JavaScript-based server-side application framework is hailed as highly scalable due to its asynchronous programming model. While small pieces of information can be served rather quickly, longer computation doesn't perform well.
- **ASP.NET/ASP.NET MVC**: Although highly popular in the enterprise sector, this option has been omitted due to non-negligible licensing cost in development (Visual Studio) and production (Windows Server), as well as vendor lock-in.

While at least as powerful as Django-Piston, Ruby on Rails is not described in this document due to limited time and resources. However, it should be mentioned that, contrary to popular belief, the performance gap<sup>11</sup> between Python and Ruby has little significance, especially in comparison to the performance of Java web services.

## 13.3 COMPARISON

### 13.3.1 PERFORMANCE TESTS

The implementations are stress-tested using Apache JMeter<sup>12</sup>, a highly configurable load testing tool. The test runs are performed using 2 threads, each sending 5000 requests. By considering the overall time taken by the application servers to handle the requests, a throughput can be computed in the unit requests per second.

#### 13.3.1.1 PROCEDURE

The application servers mentioned in section 13.2 are used to perform the test. They run on the same machine as JMeter, which has the following specs:

---

<sup>9</sup> <http://rubyonrails.org/>

<sup>10</sup> <http://nodejs.org/>

<sup>11</sup> <http://shootout.aliath.debian.org/u64q/which-programs-are-fastest.php?java=on&yarv=on&python3=on&calc=chart>

<sup>12</sup> <http://jmeter.apache.org/>

- Intel Core 2 Quad CPU, clocked at 2.66GHz
- 4 GB DDR3 RAM
- Windows 7 Professional SP1 64bit

Besides the application server and JMeter, no resource intensive tasks are running on the test machine during the tests.

In this case, a find all operation is conducted as each request to the web service. The database contains 5 sample users, which should be returned in serialized form as XML.

### 13.3.1.2 RESULTS

The test results are summarized in the following table:

	Total Time	Requests/second
<b>JAX-RS</b>	13s	783.8
<b>Spring MVC</b>	18s	565.4
<b>Django-Piston</b>	121s	82.8

The numbers above demonstrate that a JAX-RS solution performs best in terms of throughput. Spring MVC follows closely, while Django-Piston is very slow in comparison.

### 13.3.2 RESOURCE EFFICIENCY

Memory consumption was tracked during the stress test mentioned above. The following table shows the results:

	Minimum KB	Average KB	Maximum KB
<b>JAX-RS</b>	504,044	522,180	579,143
<b>Spring MVC</b>	115,288	165,037	252,770
<b>Django-Piston</b>	37,664	42,966	43,052

Django-Piston turns out to have the smallest memory footprint, while the Java-based approaches take up a lot more memory, possibly due to the overhead imposed by the Tomcat application server. The large difference between Spring and JAX-RS in favor of the former is hard to explain, since Spring should impose the larger overhead, being the more complex piece of software. An anomaly caused by server misconfiguration might have led to this outcome.

### 13.3.3 DIFFICULTY OF DEVELOPMENT

At first glance, this aspect doesn't benefit the user of the resulting product, as it seems to cater to the convenience of the developer only. However, an indirect consequence of easier development is increased developer productivity, resulting in higher quality results and more implemented features.

Due to reduced complexity, easy-to-use technologies also reduce the potential for error in the resulting software, which frees up time that would have been spent on fixing bugs for other tasks, whose results may provide a direct benefit for the end user.

That said, ease of development comes with a cost in performance, as it can be observed in the stress test results displayed in section 13.3.1.2.

### 13.3.3.1 MEASUREMENT THROUGH LINES OF CODE

As an indicator of development difficulty, the following table lists the number of lines of code (LOC) necessary to create the prototypes described in section 13.2. Note that this measurement is by no means sufficient to determine difficulty of development with certainty – it is merely a guideline. The values are approximate and neglect comments.

	JAX-RS	Spring MVC	Django-Piston
<b>Resource implementation</b>	240	220	80
<b>Data access layer</b>	380	380	0
<b>Configuration</b>	70	100	20
<b>Generated</b>	0	0	180
<b>Total hand-written</b>	690	700	100
<b>Grand total</b>	690	700	280
<b>Est. additional LOC per resource</b>	100-200	100-200	10-100

The Java-based approaches are quite similar since they share most of their data access layer. It should be also noted that a considerable portion of the Java approaches' LOC is caused by the language's lack of syntactic sugar for defining properties. However, in the case of this prototype, this reason causes overhead of no more than 100 LOC.

### 13.3.3.2 MEASUREMENT THROUGH DEPENDENCY COUNT

Another way to indicate difficulty of development is the number of dependencies. The weakness of this measurement is that the complexity of the dependency itself is not considered. The following table depicts the situation for the prototypes discussed in this document. Dependencies used directly by the prototype (including APIs and their implementation) and transitive dependencies (dependencies of direct dependencies) are differentiated. Software is counted as a dependency if it is a software package (JAR or Python Egg) that is not part of the language's (Java EE or Python 2.7) standard distribution.

	JAX-RS	Spring MVC	Django-Piston
<b>Direct dependencies</b>	14	17	2

<b>Transitive dependencies</b>	18	32	1
<b>Total dependencies</b>	32	49	3

Spring has slightly more dependencies than JAX-RS, mainly because of the framework is split up into separate JARs for increased modularity. The Django approach is more independent: Django itself has no dependencies, Django-Piston just depends on Django; the PostgreSQL driver is listed as sole transitive dependency.

### 13.3.3.3 MEASUREMENT THROUGH LEARNABILITY

As a final, somewhat subjective measurement, a brief account shall be given on the presented technologies learnability, which covers mainly their complexity and the availability of help on the Internet. The perspective is that of a developer who is proficient in Java and Python, acquainted with software patterns and some frameworks, and has at least superficially grazed Enterprise Java, Spring and Django.

JAX-RS and Spring MVC, being some of the most popular technologies to build RESTful web services, are extensively documented, and help sites contain solutions to a wide variety of issues. As it turns out, the latter is no coincidence, since setting up a JAX-RS or Spring MVC project properly is no easy feat, possibly requiring several days of slow progress for someone less experienced with those technologies. The high complexity has a noticeable impact on the effort required to gain sufficient understanding of the technologies.

The JAX-RS approach is easier learnable than the Spring MVC one, as it allows the developer to design an application more freely: In this approach, several independent technologies are tied together, while Spring MVC has a rather holistic perspective, where the application must conform strictly to guidelines imposed by the framework's architecture. This architecture's complexity makes writing a Spring MVC application comparatively hard.

Django is relatively easy to pick up, requiring 1-2 days of working through tutorials and experimenting to get a good grasp of the framework. Well written tutorials and documentation are available. Since this is a very popular framework, help for a wide range of problems is readily available as well. Django-Piston, being a very minimalistic, can be learned even quicker. However, on the downside, documentation and help for the latter is rather scarce, so one might have to take a look at Django-Piston's source code to solve more intricate problems. Over all, web services can be developed very fast using those technologies, especially since database interaction is handled entirely by Django.

## 13.4 SUMMARY AND CONCLUSION

Prototypes based on JAX-RS and Spring MVC (both running on a Tomcat application server), and a prototype based in Django-Piston were introduced in this document. All of them implement a basic RESTful user management service, which was used to investigate the qualities of the technologies.

Several other popular technologies in this area were omitted for reasons provided in section 13.2.4.

Performance tests have shown that the Java-based approaches are far superior in how many requests can be served per second: JAX-RS is more than 9 times as fast as Django-Piston, Spring MVC is a little slower, only being 6.8 times as fast as Django-Piston.

In turn, Django-Piston trumps the Java-based approaches in terms of memory footprint by factor 3.9-12, in terms of lines of code by factor 7 (handwritten)/2.5 (handwritten+generated), and in terms of dependencies by factor 10-16. The Django-Piston approach appears to be easy to learn as well. The prototype's codebase is lean thanks to a high level of abstraction, and Django taking care of all database access needs.

However, considering most developers' familiarity with the Java ecosystem (and reservations in the face of dynamically typed languages), and the Java-based approaches' superior ability to handle a large number of requests in a small timeframe, choosing one of those appears to be a better choice than Django-Piston, despite its merits.

Choosing among the Java-based approaches, JAX-RS appears to be the best candidate due to its high performance and simplicity (at least compared to the Spring MVC approach).