D2.2: WISETREP Architecture & Data Model

Due date of Deliverable: Month 8
Preparation date of Deliverable: 31/10/2008
Lead beneficiary: Forthnet
Nature: Public
Status: Final

Project Name: Wide Scale network of E-systems for Multimodal Journey Planning and Delivery of Trip Intelligent Personalised data
Acronym: WISETREP
Start date of project: 01/02/2008
Duration: 30 Months
Project no.: 213233
<table>
<thead>
<tr>
<th>Document Number</th>
<th>213233-WISETRIP-D2.2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Document Title</td>
<td>WISETRIP Architecture &amp; Data Model</td>
</tr>
<tr>
<td>Lead Beneficiary</td>
<td>Forthnet</td>
</tr>
<tr>
<td>Editor(s)</td>
<td>Vassilis Spitadakis (<a href="mailto:vspit@forthnet.gr">vspit@forthnet.gr</a>)</td>
</tr>
<tr>
<td>Work Package No.</td>
<td>2</td>
</tr>
<tr>
<td>Work Package Title</td>
<td>Requirements Analysis &amp; Architecture Design</td>
</tr>
<tr>
<td>Nature</td>
<td>Report</td>
</tr>
<tr>
<td>Preparation Date</td>
<td>31/10/2008</td>
</tr>
<tr>
<td>Dissemination Level</td>
<td>PU</td>
</tr>
</tbody>
</table>
| Revision History      | Version 0.1: 26/07/2008 (first draft)  
|                       | Version 0.2: 30/09/2008 (first complete draft)  
|                       | Version 0.3: 14/10/2008 (peer review by all partners)  
|                       | Version 0.4: 31/10/2008 (final version) |
| Contributors          | Vassilis Spitadakis (Forthnet SA)  
|                       | Maria Fostieri (Forthnet SA)  
|                       | Dimitris Bouras (Forthnet SA)  
|                       | Alma Solar (ETRA Investigación y Desarrollo SA)  
|                       | Pekka Eloranta (Mobisoft Oy)  
|                       | John Nelson (University of Aberdeen)  
|                       | Paulus Aditjandra (University of Aberdeen)  
|                       | Eleonora Ercoli (MemEx srl Livorno)  
|                       | Hanne Lindqvist Rantala (Destia Tieliikelaitos) |
Table of contents

1. INTRODUCTION.................................................................6
  1.1 Overview .................................................................6
  1.2 Methodology.............................................................6
  1.3 Features of the WISETRIPT system.........................7
2. REFERENCE ARCHITECTURE..................................................9
  2.1 Overall Architecture..................................................9
  2.2 Participating Services And Source Data ..................10
  2.3 Main System .............................................................11
    2.3.1 WISETRI Core System........................................11
    2.3.2 WISETRI Personalisation System..........................12
    2.3.3 WISETRI Services...............................................12
    2.3.4 Language Adaptation..........................................12
    2.3.5 Interfaces to devices..........................................13
  2.4 Distribution & Presentation.........................................13
3. DESCRIPTION OF THE WISETRIPT ARCHITECTURE...............15
  3.1 Functional Architecture of the Main System..............15
    3.1.1 The Main Problem ............................................15
    3.1.2 Core Functions of WISETRI.................................18
    3.1.2.1 Algorithmic Thoughts ...................................23
    3.1.3 WISETRI Personalisation Functions.....................26
    3.1.3.1 Real-Time Decision Functions.........................26
    3.1.3.2 Personal Trip Life Cycle Management Functions....30
    3.1.4 WISETRI Services Functions...............................32
    3.1.5 Language Adaptation Function................................33
  3.2 WISETRI External Systems ...........................................34
    3.2.1.1 Participating Journey Planners functionality ..........34
    3.2.1.2 Booking & Payment functionality......................34
    3.2.1.3 Other data providers......................................36
  3.3 Distribution & Presentation Functionality..................37
    3.3.1 Fixed Web Functionality......................................37
    3.3.1.1 General Web Functionality................................37
    3.3.1.2 Personalised Web Functionality.......................47
    3.3.2 Mobile Functions...............................................49
    3.3.2.1 General Mobile Functionality............................50
    3.3.2.2 Personalised Mobile Functionality....................51
    3.3.3 Functional Implementation Considerations...............51
  3.4 Software Communications Architecture.....................52
    3.4.1 Interfaces to Participating Journey Planners..........52
    3.4.2 Interfaces to devices.........................................54
    3.4.3 Database Communications.....................................55
    3.4.4 Communication interface to other external systems.....55
    3.4.5 Software Communications Implementation Considerations ...56
4. DATA FRAMEWORK ..........................................................57
  4.1 Data Architecture.......................................................57
    4.1.1 Data about Journey Planners – Metadata .................57
    4.1.2 Journey Planner Specific data..............................59
    4.1.3 User Profile Data................................................60
4.1.4 Personal Trip Cycle Data................................................................. 61
4.1.5 Trip Life Cycle Data.................................................................... 62
4.1.6 Linguistic Data ............................................................................ 64
4.1.7 Other Data – User Context – Location based............................ 65
4.1.8 Data Schema and Format Considerations................................. 66
5. PHYSICAL ARCHITECTURE ........................................................................ 68
  5.1 Servers.......................................................................................... 68
  5.2 Databases ...................................................................................... 69
  5.3 Network .......................................................................................... 69
  5.4 Devices .......................................................................................... 69
  5.5 Implementation Considerations..................................................... 70
6. FINDINGS ON RISKS AND CONSTRAINTS.................................................. 71
7. WISETRIP GLOSSARY............................................................................. 72

Table of figures
Figure 1: Methodological tasks for the definition of the architecture.......... 7
Figure 2: Overall WISETRIP Architecture............................................... 10
Figure 3: The Main Problem .................................................................. 16
Figure 4: Core System flow diagram ..................................................... 23
Figure 5: A query example ..................................................................... 25
Figure 6: Personalisation System flow diagrams.................................... 29
Figure 7: Booking and Payment functionality........................................ 35
Figure 8: Private car trip / journey and potential data implications provided by JPs
.................................................................................................................. Errore. Il segnalibro non è definito.
Figure 9: Public transport trip/journey and potential data implications provided by JPs
.................................................................................................................. Errore. Il segnalibro non è definito.

List of tables
Table 1: Core Functions of WISETRIP .................................................... 19
Table 2: The two different algorithmic approaches ................................ 26
Table 3: Personalisation functions.......................................................... 26
Table 4: Language Adaptation functions .............................................. 33
Table 5: Relevant data provided by JPs .................................................. 36
Table 6: User Profile Data...................................................................... 60
Table 7: Personal Trip Life Cycle Data .................................................... 62
Table 8: Trip Life Cycle Information...................................................... 64
Summary

WISETRIP project is a Collaborative project co-financed by EU FP7 programme under Theme TPT.2007.4 “The connected traveller in the city, region and world of tomorrow”. The goal of WISETRIP is to develop and validate an innovative mobility service platform, which provides personalized multi-modal trip information sourced from connected variant journey planners. The proposed system is accessible by travelers through various mobile or fixed terminals/devices before and during the journey. The project will take the reference of the existing independent systems for journey planning that will become active sub-systems of a ‘global’ journey planner system.

Task 2.3 – “Architecture Design and Data Framework” aims to specify the WISETRIP system from a functional and architectural standpoint. To this end, this Task will: a) establish a reference model architecture that will describe the way the WISETRIP journey planner (Core System) will be composed from other journey planning systems, the communication layers, the function role of the decision mechanism & adaptation modules that are involved (trip life-cycle manager, personalisation data & user profiles and information adaptation and distribution gateway), and b) identify the general methodologies, concepts, data architecture and data requirements, functional modules and ideas that will be explored in the following workpackages to pursue the specific technical and scientific objectives of the WISETRIP project, as well as the tools and technologies that will be used to build up a fully functional prototype of the WISETRIP system.

The outcome of this task is the present Deliverable, which defines the system model that will be used as basis for all the design and implementation activities in WP3 (Journey Planner – Core System Development), WP4 (Personalisation System Design & Development) and WP5 (Information Distribution System Design & Development), as well as during the trial phase (WP6).
1. INTRODUCTION

1.1 Overview

The objective of this Deliverable is to describe in a concrete way the overall architecture of the WISETRIP system, the necessary modules, their functional role, the communications messaging, the data architecture and data requirements. It is based on the outcomes of Task 2.1, which identifies the requirements of the users of the WISETRIP system, as well as the outcomes of Task 2.2, which identifies how the WISETRIP system should interact with the participating Journey Planners and how it should function in order to accommodate the potential non-uniformities of the participating systems in terms of performance, functional differences, data representations.

The remainder of this report consists of 5 sections:

- Section two presents the reference architecture, where the overall architecture of the system is described along with the system’s three fundamental layers (the Participating Services and Source Data Layer, the Main System and the Distribution Layer)
- Section three presents in details the functional architecture of the above layers as well as the communication links that need to be established
- Section four presents the data framework, where we identify the required data for each functioning module to perform
- Section five describes the physical modules that will be utilized i.e. servers, databases, devices, network issues etc. Finally,
- Section six presents major findings on risks and constraints that need to be addressed.

1.2 Methodology

In order to define the architecture of WISETRIP in a way that it will be complete and easy to be understood within the development phase, the following levels of architecture will be analysed:

a. Functional Architecture: Here, the operation of the desired system is decomposed into layers and functions, based on the study of the requirements analysis of WISETRIP. Each function is analysed in terms of its necessary input, its output and its interaction with other functions. Finally, a diagram can represent the overall synthesis of functions and roles into the complete functional architecture. The functional architecture and its analysis is the basis in which further architecture definition (related to data and communications) will be made. User interaction which will be realized from the user interface functions are also a special part of the functional architecture.

b. Data Architecture: Having completed the functional architecture, a thorough scanning of each function is described in order to identify the required data for each functioning module to perform. Through this process, a global view of the required data is obtained. Gathering of data and analysis of interrelated data and their use, leads to the identification of main data entities required. Data architecture is not getting into the detailed level of a database design, but brings analysis up to the identification of the information entities that need to be represented, their main attributes and the main relations among them.

c. Communications Architecture: Apart from the actual required data, functions might be implemented by functional entities (software procedures, algorithmic modules, etc) that will need to interact among each other even operating at different physical infrastructure. Moreover, functions might interact and exchange
data with external systems, not controlled by WISETRIP system. In both cases, communication mechanisms will be necessary and will be established. Communications architecture describes these communication links and attempts to analyse the way and technological method to perform the communication exchange as well as the necessary fields and data components that are exchanged.

d. **Physical Architecture:** Description of physical modules that will comprise the solution. Physical modules include the tangible equipment that will be utilized i.e. servers, computers, network equipment, network links and topologies etc.

The detailed definition of the above levels (functional, data, communications and physical architecture), resulted by employing the following methodological tasks, which run in parallel or sequentially as described in the related schema (Figure 1):

1. Study User Requirements
2. Problem Analysis
3. Initial Concept for the Reference Architecture
4. Definition of Glossary
5. Identification of main functions
6. Analysis of Functional Architecture
7. User Interface Architecture
8. Identification of Data Requirements
9. Data Architecture Analysis
10. Communication Architecture
11. Physical Architecture Analysis
12. Revision (of Reference & Detailed Architecture)

Though not a necessary component of the architecture, there is a final task, identifying challenging issues and risks related to the implementation of the architecture. Through this task algorithmic considerations are made and the implementation risks are identified.

![Figure 1: Methodological tasks for the definition of the architecture](image)

**1.3 Features of the WISETRIP system**

The architecture should ensure proper functional behavior of the system and data efficiency so as to reach the project objectives. Towards this achievement, the key features envisaged will be:
Being a one-stop-shop for journey planning at wider scale achieved through effective cooperation among existing independent systems for journey planning that will become active subsystems of WISETRIP. Final system will combine different answers from different systems that might not even be known to the users.

- Openness and ability to incorporate well-documented and specified interfacing modules so that new journey planners can embrace the global system and become active components that expand its geographical coverage to new cities, regions or nations.

- Integration and algorithmic management of incorporating graph of journey planners, as well as the synchronization needs between the real-time change of journey parameters and the traveller's needs within the trip process to strengthen the impact of providing personalized information.

- Processing of information to achieve high level of personalisation of the final information delivered to the end user. User-friendly, user-empowered and efficient information will be the key towards a seamless journey planning service to the end user. Provision of instant information requires increasing system complexity and will use the personalization engine of WISETRIP based on multiple personal criteria either defined before the trip or based on real-time data and events.

- Adaptation of presentable information to multiple devices and presentation means.

- Enhanced journey-planning capability, by developing a service for planning multi-modal journeys beyond the borders of European countries or regions by taking into account multiple criteria and complex scheduling constraints (i.e. time windows on departure or arrival) applied to a network of variant journey planners.
2. REFERENCE ARCHITECTURE

2.1 Overall Architecture

The envisaged system can be divided into three fundamental layers. These are:

1. The Participating Services and Source Data Layer
2. The Main System
3. The Distribution Layer

They are described briefly below (from bottom to top) with subsequent sections make a more detailed analysis and are finally elaborated into the functional and data analysis described within this deliverable:

- **1st Layer - Participating Services and Source Data**: These are the existing systems (such as journey planners or booking systems), external information and services (such as time, user position, external alerts) as well as the data environment including personal user data related to trip selection and user preferences.

- **2nd Layer – Main System**: Here we identify three main modules, the journey planning (WISETRIP Core System), the personalization system and, on top of them the services. The final journey plan is produced by the Core System and is handed to the Services for delivery or to the personalisation system for further processing. Other sub-systems refer to interfaces with the other layers or data sub-systems. The WISETRIP Platform is interfacing to the 1st layer: a) The Core system interacts with the participating Journey Planners and b) the Personalisation System interfaces with external information, user data and profile through the real-time decision mechanism, which process all external & user data. The Services of the WISETRIP Platform are communicating with the devices of WISETRIP (3rd layer) through an intermediate layer which is ‘Devices Interfaces Layer’ that takes care of the open & harmonised interconnection to various devices.

Knowledge of the platform includes a) data (geography, destinations, means, and terms) about participating transport networks (that are presented through the participating journey planners of the 1st layer) and b) trip cycle model that is being used in order to build the personal trip life cycle data of each traveler (1st Layer)

- **3rd Layer – Distribution**: We call ‘distribution’ the function of providing the information and service to multiple users through various technological devices. The main role here belongs to the devices and the user interfaces running on top of them.

Subsequent paragraphs overview all components of the three layers which are illustrated in Figure 2 below:
2.2 Participating Services And Source Data

In the Participating Services and Source Data layer of WISETRIP architecture we foresee all the pre-existing systems (i.e. journey planners or booking systems), as well as other external information sources and services and the necessary data which might include personal user data related to trip selection and user preferences. A brief introductory description of these functional and data modules is given below:

**Participating Journey Planners** - The WISETRIP journey planner platform is connected to many Journey Planners of variant type, which are serving different or overlapping areas but at the same time are having advanced control capabilities according to adaptations made to them through the activation of the necessary interface. These are the participating Journey Planners, each one controlled by a different transport operator or content provider, but conforming to WISETRIP adaptation interface guidelines so that they can function and respond under the umbrella of the WISETRIP wide-scale Journey Planner Platform. A participating Journey Planner can be either an Urban Journey Planner, or a Long Distance Journey Planner or even a national/regional Journey Planner, including both urban and long distance journey planning services. WISETRIP focuses on the protocols and software mechanisms, which will enable a combined use of many interconnected journey planners, through a single user interface.

**User Profiling** - The provision mechanism for the necessary personal information for a selected trip takes into account the configuration of users’ profile (username – password, e-mail or mobile phone number where alerting messages will be forwarded by the system, frequent questions, early trip messages, selected trips, etc.), that will determine the activation of the appropriate Device Adaptor/Gateways in order to provide to user’s device access to the system, and allow the Trip Life Cycle Manager to specify when and which information corresponds to pre-defined user preferences.

**Personal Trip Data (or Trips Life Cycle data)** - Each journey is scheduled according to Users Profile and criteria of Trips Cycle selected by the user. Every trip is registered
into a user profile and a specific trip life cycle for a specific user, based on the applied trip life cycle model, is created. Specific date(s) and time(s) for each phase are stored as well as predefined messaging types and templates that will be followed by the personalisation system.

**External Data Sources** - For the provision of valid integrated traveller information services, the availability of external data sources (e.g. ports, airports, train stations, etc.) is of major importance. The External Data Wrapper system will be the middleware software, through which the Personalisation System (Real-time decision mechanism or Trip Life Cycle Manager) access heterogeneous data sources and receives interesting messages i.e. a weather alert, or a trip cancellation alert. The External Data Wrapper will ensure the communication with every different external system and will provide a common API in order to access heterogeneous systems with information of the same type.

**Booking & Payment Systems** - The Booking and Payment system is an external system that might or might not exist for a certain trip segment. It is the booking engine that implements, for selected routes specified by the Journey Planner, ticket issuing and payment for user journey. Some components of the final user response (i.e. one or more trip segments) could link to booking systems for the ticket purchase and on-line payment functions.

### 2.3 Main System

The WISETRIP Main System consists of three main modules, the journey planning (WISETRIP Core System), the personalization system and, on top of them, the services. The journey plan response is produced by the Core System and is handed to the Services for delivery or to the Personalisation System for further processing before its delivery to the end user. Other sub-systems refer to interfaces with the other layers or data sub-systems. The Main System is interfacing to the 1st layer, as: a) The Core system interacts with the participating Journey Planners and b) the Personalisation System interfaces with external information, user data and profile through the real-time decision mechanism, which processes all external & user data. The Services of the WISETRIP Platform are communicating with the ‘WISETRIP Devices’ (3rd layer) through an intermediate layer which is the ‘Devices Interfaces Layer’ that takes care for the open & harmonised interconnection to various devices.

Knowledge of the platform includes a) data about participating transport networks and the journey planners and b) trip cycle model data that are used in order to build the personal trip life cycle data of each traveller. All these entities and terms are described within the next sub-sections where the functional architecture analysis and analysis of data framework goes into more detail.

#### 2.3.1 WISETRIP Core System

The Core System of WISETRIP is the sub-system that realises the mechanism through which the necessary information coalition and services interconnection are implemented. The search for the optimum routes and journey planning will be executed by the Journey Planner application that will implement the search for transportation routes the user prefers. Henceforth, core system functions include the main algorithmic solutions which will find out which external participating Journey Planner to communicate with, how and what to ask, and then compose the final response that will be presented to the user/traveler.

The Core System will need to utilise:

- **The Journey Planner Data**: data necessary for the knowledge about the journey planners. This knowledge could consist of destinations names, technical details (port numbers), capabilities, identifiers of interface components, URLs, addresses, means/modes of transport, etc.
2.3.2 WISETRIP Personalisation System

The WISETRIP Personalisation System performs journey planning functions and procedures according to the users profile and trips cycle criteria selected by the user. It detects the information that must be sent to each user, processes alerting messages and produces notifications, defines when and how to send a notification message, and utilizes the necessary mechanisms in order to distribute this information.

Real-Time Decision Mechanism - An internal module of the WISETRIP Personalisation system is the Real Time Decision Mechanism which mainly constitutes event listener applications or continuous monitoring processes, that receive/produce real-time information on a permanent (or on per request) basis and then trigger decision mechanisms that will define what information will be ‘pushed’ to interested users (either a location based service, or an event that must be disseminated to registered users). It communicates with the external data sources and receives information by them (information push). Every piece of information is analysed and according to the trip cycles that are in progress it triggers one or more events for one or more users.

Trip-Life Cycle Model - The Real Time Decision Mechanism also takes into account pre-registered trips selected by the users, and specifically the different periods of a journey (trip – specific life cycle data) that constitute the Trip Life Cycle Model (Pre-trip phase, En-route phase and other phases of a journey). The trip life cycle model is defining the rules on which journey analysis will be made into phases, so that a specific user journey is translated to Trip Life Cycle data. The decision mechanism sets up a schedule of notifications into time for a specific trip, when relevant up-to-date information is ‘pushed’ to each user. Information is analysed and according to the trips cycle sets off one or more notifications for one or more users. The Trip Life Cycle manager is identified as a separate component, part of the real-time decision mechanism, and also includes functions related to the definition of personalised data (profile definition function, trip selection, etc.).

2.3.3 WISETRIP Services

All the functions implementing the actual WISETRIP services are included in this module and are gathered together, they define the set of end user services on which the final interface will be based on. WISETRIP services can be divided to:

- **Passive services**: referring to services that are provided to a user on demand (Information Pull). It is the case where a user through his device requests information by the WISETRIP system. Passive services will be accessed by third-party applications through an Application Programming Interface (API) - the WISETRIP Services API.

- **Active services**: referring to services that are automatically provided to the user according to specific events occurring during the progress of a trip’s life cycle (Information Push). The Active services receive information about these events either by the Real-Time Decision Mechanism or by external data sources utilizing the WISETRIP Information Reception API to dispatch that information.

2.3.4 Language Adaptation

It is obvious that for a system aiming to be an integrator system of international level journey planners it is prominent that all information and all functions / services should be supplied in multiple languages. To this end, the WISETRIP system has to ensure the translation of content and services at least in the European languages of consortium
participants and to implement the mechanism that will detect from user information his/her preferred language and dynamically adapt in it.

The Language Adaptation mechanism will implement the following functions:

- Translations of static content and data of the core system where required
- Implementation of dynamic translation basically for the essential parts of the information which are the system responses & notification messages
- Automated language selection (or recommendation) based on user information or the origin of the request.

### 2.3.5 Interfaces to devices

Since the WISETRIP services must be accessible through different user devices, the passive system that implements these services will provide access methods through a common API. In order for the system services to be constantly accessible by a specific device a middle application module (Device Gateway) adapting the communication between a device and the WISETRIP system will be developed. The Device Gateway could consist of a set of functions that in a generic way can interface with several device applications / interfaces.

Through the implementation of Device Gateways the system will be easily expandable to be utilised by new user devices. In order for a new device to access the WISETRIP system, the development of the relevant Device Gateway will be necessary, without modifications of the Core System to be required. In addition, a Device Gateway may be developed by anyone that is not aware of the WISETRIP Core System, but is only aware of the WISETRIP Services API.

A Device Gateway can be an interface to SMS messaging server or to a mail application that takes over to deliver mail notification to users when it is necessary.

### 2.4 Distribution & Presentation

The objective of WISETRIP is to develop the necessary interfacing mechanisms so that the realised services can be distributed and presented to the end users through fixed or mobile devices connected to the Internet, before and during the journey. The aim is to represent a user-friendly and easy-to-use combined transportation information system, which accompanies the traveller/user starting from the search of a route and the time of his departure, until the arrival to the final destination.

The traveller/user of WISETRIP has access to this information through multiple interfaces and specifically through a fixed web interface (web portal), a mobile interface (mobile device or PDA & mobile web-browser), and a messaging interface (mobile device or PDA & SMS).

A brief introductory description of main user interface modules is given below:

**Fixed Web Interface** - Access to the WISETRIP services through computers or information kiosks is provided through a public Web Portal interface utilizing an appropriate web browser. The web portal provides the users of the system with a graphical user interface for the WISETRIP services offered.

The input composition of the fixed web interface consists of query synthesis and user options, etc, while the output presentation demonstrates the relevant (personalized) results, post-processing, filtering, etc. The interface also includes a user-profile editing section in order to provide the user with the chance to modify his profile at any time as well as a trip registration screen. In addition, the fixed web interface incorporates a private personalised section where the user can indicate his journey specific preferences.

A separate administration interface will be necessary in order to manage the WISETRIP data.
Mobile Interface - Through the mobile interface a mobile version of the public web portal is offered to the users. A special personalised interface is developed for location-based services.

Additionally, the mobile interface incorporates presentation functions for SMS messages that are forwarded to the user according to the schedule created for a specific journey. The messages relevant to the user’s journey are dispatched through the messaging interface.

Messaging Interface - The messaging interface undertakes the presentation of user specific messages (alerts, notifications, and scheduled messages concerning a journey) through the available means and supplements the distribution / presentation layer of the WISETRIP system architecture.

Based on the schedule created for every journey, the relevant messages (by SMS or E-mail according to the users profile), are distributed to each user related to them through the utilization of the messaging interface. Special messages concerning any changes and/or modifications to a journey are directly forwarded to the relevant users regardless of the journey schedule.

An SMS Gateway functioning under a client-server model will be utilized for the distribution of SMS messages. Access to the SMS Gateway is provided through the TCP/IP connections and the utilization of a specialised protocol based on the XML standard.

Through the same client-server model, an e-mail server is utilized, which is responsible for the distribution of e-mail messages.
3. DESCRIPTION OF THE WISETRIP ARCHITECTURE

3.1 Functional Architecture of the Main System

Functional Architecture analyses the roles of software components of variant size, type and complexity; and the relations among them in terms of input and output dependencies as well as ordering in time. Flow diagrams and textual descriptions will be utilised in order to make the architecture understood to the subsequent stages of the project.

Before getting into the description of functions, the main problem addressed is thoroughly described, so that a common base can be shared in the remaining analysis. Then, the Core Functions of the system are described, using a tabular and graphical way (flow diagram). A series of thoughts concerning algorithmic solutions is made for specific functions, addressing important implementation issues that are strongly dependent on the functional choices made and affect also data requirement. Personalisation functions are also analysed and described, separated into those related to real-time decision mechanisms and those related to the management of personal trip life cycle information. Finally, the services layer that provides the means to realize and put functions into action, are described.

Moreover, peripheral layer functions are also described, such as language adaptation, booking and payment and external functions, and functions related to presentation for the fixed and the mobile scenarios.

Analysis of the functional architecture ends up with a statement mentioning implementation considerations derived from the functional architecture described so far.

3.1.1 The Main Problem

To understand the functional architecture and before getting into its analytical description, the main reference problem should be described as well as the expected system response.

The WISETRIP main problem is more or less similar to that of any typical journey planner. The user is submitting a query, asking for available transport solutions offered by various alternative means of transport, from a specific origin location A to a specific destination location B. Time co-ordinates of the desired move, as well as user preferences (most of them are optional) such as preferred means of transport, maximum duration of the journey, maximum number of hops, are also among the main elements of the query at the user level.

Ideally, WISETRIP – and any Journey Planning system – should extract from its knowledge base and its co-operative systems, a list of trip solutions, which compose the response of the system and should include all available ways to commit the trip that fit to the specific user needs.
The particularity of WISETRIP stems from the fact that the knowledge about trips and consequently about their potential combination into a feasible journey from the Origin A to Destination B, is not available within the dataset of WISETRIP, nor can it be obtained through processing other transport operator’s data, but should be obtained through cooperation with other Journey Planning systems that cover, from the transport network point of view, these locations. Henceforth, final response should be based upon results delivered from participating systems, which are actually Journey Planners covering specific areas and transport networks, and for which WISETRIP must find algorithmic ways to establish communication with them and compose results for the end user request, including as much analytical information as possible for each stage of the journey.

Figure 3 above illustrates the user level aspect of the main problem, as well as some specific internally identified entities, for which definitions will be made and will be used hereon within this document:

**USER QUERY:** It is the initial request of the user for trip information for a specific journey. User Query is composed by the main Origin-Destination pair (A and B in the Figure above), along with special criteria that limit the number of possible answers and make the query more specific and personal, based on the needs of the traveler. Main potential components include:

- Origin (country, city, area, street, other)
- Destination (country, city, area, street, other)
- Departure Time (approximately, before, after)
- Arrival Time (approximately, before, after)
- Duration of the journey (exactly, approximately, greater than, less than)
- Number of Hops (or limit of Hops)
- Modes of Transport and preferred ones (air, rail, ferry, bus/coach, metro, ...etc, including walking mode)
- Preferred intermediate nodes
Cost Limit
Number of Persons
Sorting Preferences (by duration, by means of transport, by departure time, by arrival time...)
Special Needs and Requirements i.e. E&D, vegetarian, green routes

**JOURNEY or JOURNEY SOLUTION:** It is the description of the total travel requirements from start at origin location to finish at destination location. It may comprise one or more trips using one or more modes.

**HOP or TRIP:** It is the individual movement between two location points.

**QUERY RESPONSE:** It is the list of journey solutions that satisfy the user requirements, and commit the movement from Origin A to Destination B, within the time requirements of the user and according to any other special requirement. Each journey can involve more than one transport mode and can be represented as a sequence of hop trips. Full analysis of each journey includes all details for each hop.

**SEGMENT or PARTIAL JOURNEY:** A segment is a subsequence of trips that represent a part of a journey and correspond to a specific move from an intermediate location (or even the Origin location A) to another intermediate location (it could be the destination location B)

**INTERCHANGE POINT:** An intermediate location within a trip, where the user should disembark from a specific means of transport and get on board other means of transport, either of the same mode (i.e. a railway station where he should change train) or different mode (i.e. a terminal rail station at a port site, where the user is asked to move from the rail to a ferry).

**ORIGIN JOURNEY PLANNER DOMAIN:** It is the set of segments that result from a single Journey Planner and correspond to partial journeys starting from the Origin location, reaching up to an intermediate location (i.e. Location X in our figure above)

**DESTINATION JOURNEY PLANNER DOMAINS:** It is the set of segments that result from a single Journey Planner and correspond to partial journeys reaching the Destination location, starting from an intermediate location (i.e. Location Y in our figure above)

**EXTRA JOURNEY PLANNER SEGMENT:** It is the segment, linking two intermediate locations of the journey, which however has no common hops with the Origin Journey Planner Domain and the Destination Journey Planner Domain.

**INTERNATIONAL SEGMENT:** Usually, and from the practical point of view, the Extra Journey Planner Segment represents an international journey that is necessary within the total journey solutions that are available for the user. This happens because, most of the journey planners find routes within a single nation, though in the WISETRIP cases, we explored the needs of the international traveler.

**Example:** In Figure 3 we note the following:

One Journey Planner provides results for the segment “from origin A to X” and the other Journey Planner provides results for the segment “from Y to destination B”. We call them,

the Origin Journey Planner (JP) Domain which contains the following segments:

- Segment One: A-I3-X
- Segment Two: A-I1-I2-X
- Segment Three: A-I4-I5-I6-I7-X

and the Destination JP Domain which contains the following segments:

- Segment Four: Y-K1-B
Moreover, the two sets of segments can be linked through segment “from X to Y” which is considered as an extra Journey Planner segment, and can be as follows:

- Segment Six: X-Y
- Segment Seven: X-M1-Y

Combining all the above segments, we can produce feasible journeys “from A to B”. However, we can limit them when we apply specific time and user requirements. So, WISETRIP could possibly include only the following journeys into the final Query Response:

- Journey One: A-I3-X-Y-K1-B (concatenation of Segments One, Six and Four)
- Journey Two: A-I1-I2-X-M1-Y-K1-B (concatenation of Segments Two, Seven and Four)
- Journey Three: A-I3-X-M1-Y-K2-B (concatenation of Segments One, Seven and Five)

### 3.1.2 Core Functions of WISETRIP

The Core Functions of WISETRIP aim to implement the basic service, which is the submission and processing of a specific User Query for trip information, as well as the provision of final response that is expected to satisfy the user demand according to his/her specific criteria and preferences. The Core functions are interacting with the Journey Planner interface in order to get the necessary information from the end sources of information which are the participating Journey Planners. To this aim the WISETRIP Core System includes the following functionality:

1. **User Query submission:**
   - **Initial User Query** (function 1): The system takes as input the origin and destination, the time constraints and other limitations or user preferences that might exist and responds with one or more feasible trip routes.
   - **Validation of User Query** (function 2): Before processing the user query, the system validates the query data (i.e. location names) and if needed user is asked to re-edit the query.

2. **Trip segmentation to partial journeys:**
   - **Analysis to partial journeys**: As mentioned above the user asks for a trip between two nodes and the system will utilise participating Journey Planners (each covering different areas) to respond to the user request. That means that the system might need to ask several JPs to respond to user query. Consequently the system identifies whether the initial User Query involves answers from one or more JPs (function 5), analyzes the requested trip to partial segments and formulates a series of queries to JPs (function 7) that will provide the partial answers giving the trip segment to examine.
   - **Management of international segments**: User query might include an international segment predefined by the user (function 3) or not meaning that the system should also take in mind the appropriate international route. In case that the international segment is not predefined by the user and since the participating in WISETRIP JPs do not cover the international domain the system should know basic information about feasibility of transport among international end points to handle such user queries (functions 4 and 6).

3. **Formation of final response**
   - **Searching routes for each trip segment**: through the Journey Planner Interface, a sub-system of the WISETRIP Core System, each partial routing problem is forwarded to the appropriate participating JP (function 8) which replies
with one or more optimum routes (function 9). An important issue is the order that the partial routing problems are forwarded to the JPs. It is reasonable if there are international segments to be answered first since international trips are more tight, and then based on these international routes the time constraints should be adapted to national routes where things are more flexible (because of more available transportation means and more frequent routes).

- **Combination of trip responses**: Given all partial responses from the JPs as well as the international segment, the system produces complete trip routes (function 11) and presents them to the user (function 13).

- **Responses filtering** (function 12): Some of the available responses might not fit the user's criteria and these are marked as filtered out.

The Functional Architecture of WISETRIP Core System should assure the functions listed Table 1 below. Where appropriate, one or more functions will be assigned to a software module that is named within the table.

### Table 1: Core Functions of WISETRIP

<table>
<thead>
<tr>
<th>Nr</th>
<th>Title &amp; Brief Description</th>
<th>Name of module to implement the function</th>
<th>Data Required / Other Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Form the user query.</td>
<td>Module: Query Form</td>
<td>Query Format and Query Data (i.e. main From: and To: nodes lists)</td>
</tr>
<tr>
<td></td>
<td>The user through the user interface inputs his options for the request to the WISETRIP system, actually he describes the trip information he is searching for. System takes this input and forms in a certain format the WISETRIP User Query, which is the initiating data for any user scenario that WISETRIP supports. The query data can include Origin, Destination and criteria such time of departure, arrival time, duration, number of hops, limit on the number of hops, preferred means of transport, cost limit, number of persons, sorting criteria, etc.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Validate user query data</td>
<td>Function interacts with JP Interface</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Part of the query data needs to be validated i.e. a location name, by an external system, in our case, by a Journey Planner. This function forwards such a validation request to a Journey Planner and returns either a positive response, or a negative response along with suggestions for the right information.¹</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Provide a predefined segment</td>
<td>Function interacts with User Interface</td>
<td></td>
</tr>
<tr>
<td></td>
<td>User specifies the exact details of a given trip that is the main segment of the trip. Usually, such segment will be the international (or other) trip of the user. In this special case the user has already decided the main segment of his/her trip, which in most cases corresponds to the international segment of his/her trip.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

¹ Validation also offers the mean to deal with spatial and naming proximity issues at the WISETRIP level; the issue has been already raised within D.2.1 Report on the Analysis of User Requirements
<p>| <strong>4</strong> | <strong>Get international (or extra JP) segment(s)</strong> |
|       | This function is optional, as the user might request the system to find the segment of the trip either automatically or after specifying the origin and destination nodes of the trip. |
|       | The user queries the system, by giving specific segment details (origin, destination, time preferences) and the system decides which system to interrogate and finally presents available segments. From these results, the user can select one or more segments that will be investigated and taken into account into the final synthesis of the WISETRIP response. This function is optional, as the user might request the system to find the international (or extra JP) route automatically or the segment is predefined (function 2). |
|       | <strong>JP data to provide information related to which JP to contact for the segment</strong> |
|       | <strong>Function interacts with</strong> |
|       | - Query Forwarder in order to interact with the appropriate JP |
|       | - User Interface where the user selects one or more from the proposed segments. |
|       | <strong>- Extra JP/International Query Form, which provides the query details instead of the user</strong> |
| <strong>5</strong> | <strong>Identify JPs</strong> |
|       | This is the function that identifies |
|       | a. whether the query involves answers from one or more JPs and |
|       | b. which JPs will be asked and |
|       | c. whether international trip segment is necessary (if not given) |
|       | The function is also invoked when the User query is formed, in order to initiate the validation function as described earlier for parts of the query data. |
|       | <strong>Name of Module: JP Finder</strong> |
|       | <strong>Data required will include JP Data to provide main information about the coverage of the JPs (region, country, means), the international (terminal or outbound) nodes covered and extra JP/international network information.</strong> |
| <strong>6</strong> | <strong>Formulate (automatically) query for International (or extra JP) Route</strong> |
|       | Here, the system identifies automatically that it must issue certain queries to find segments that are not falling within the participating JPs domains i.e. international segments. The algorithm engaged here, should find the query (one or more) details and forms the final queries. To do this, the system should know in advance basic information about feasibility of transport among international end points. |
|       | <strong>Module: Extra JP/international Query Form</strong> |
|       | <strong>Data required will include JP Data to provide main information about the coverage of the JPs (region, country, means), the international nodes covered and extra JP/international network information.</strong> |
|       | <strong>Function interacts with the module Extra JP/international Segment(s) Finder</strong> |</p>
<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
</table>
| 7 | **Produce JP queries for segments.**  
Formulate series of queries to JPs that will provide the partial answers giving the trip segment to examine. Many algorithmic solutions might be employed here. Decision should also take into account the international segment(s) that are selected and are feasible within the criteria set of the user (according to the user query).  
**Module:** User Query Segmentation |
| 8 | **Forward a JP segment query**  
System has defined the elements of a single query to a single JP, and it formulates it into the necessary format and forwards it to the JP through the JP interface.  
**Module:** Query Forwarded |
| 9 | **Receive a segment JP query result.**  
System waits for the JP to respond, after a JP segment query has been forwarded to it, and it receives the response which is the segment JP query result for the given query. The response is aligned to the format that the JP interface conveys.  
**Module:** Query Listener |
| 10 | **Adapt segment JP response**  
Normally, all JP responses should be compliant to generic WISETRIP interface format that the JP interface should support. However, in general, an adaptation function might be necessary in specific cases where data set in the response is incomplete and should be filled, or formatting of data needs to be adapted appropriately.  
An adaptation function should also unify the responses from different JPs. Unification should take into consideration for example time zones used by the different JPs, national co-ordinate systems used (in case of locations to be included in the response) and other data types that might need conversion in order to be filtered in the latter steps of the core functionality.  
**Note:** In principle adaptation should take place at JP level and not at the Core functional level of WISETRIP. The definition of JP interface, and its adoption by JPs, will determine at what extent this function will be necessary, and if it is, what exactly should perform.  
**Module:** JP Response Adaptor |
| 11 | **Combine segments into responses**  
Given all segment JP responses, as well as the international or extra-JP segments, this function performs the combination of these segments into complete trip routes.  
**Name of Module:** Segment Combine Module |
<table>
<thead>
<tr>
<th></th>
<th>Filter list of responses.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Based on the user criteria, responses that do not fit are omitted. If response is null then omitted answers might be useful. For this reason, these answers are not dropped out of the response, but just marked as filtered out, though possible alternatives to the user if he is to modify his criteria.</td>
</tr>
<tr>
<td></td>
<td><strong>Name of Module: Response Filter</strong></td>
</tr>
<tr>
<td></td>
<td>Data required will include user profile data</td>
</tr>
<tr>
<td></td>
<td>Present response</td>
</tr>
<tr>
<td></td>
<td>The response data have been produced, and are ready to be presented. The system passes the data to the user interface layer in order to present them to the end user.</td>
</tr>
<tr>
<td></td>
<td><strong>Name of Module: Final Response Form</strong></td>
</tr>
<tr>
<td></td>
<td>Function interacts with User Interface</td>
</tr>
</tbody>
</table>

All modules mentioned in the table above along with their functionality and interactions are presented in the following flow diagram.
User defines the trip info he/she wants. The initial query might include a predefined segment for an international trip, or user might request from the system to find an international route. System might also automatically identify the need to issue an international trip query. System validates queries and identifies if and which JPs need to be asked. If the query is not valid user is informed and asked to reedit the query.

System formulates series of queries to JPs that will provide the partial answers giving the trip segment to examine.

System formats and forwards a single query to a single JP.

In cases where data set in the response is incomplete and should be filled, or formatting of data needs to be adapted this adaptation function is necessary.

System combines all segment JP responses into complete trip routes.

Based on user criteria, responses that do not fit are marked as filtered out.

System passes the data to the user interface layer in order to present them to the user.

Figure 4: Core System flow diagram

### 3.1.2.1 Algorithmic Thoughts

It is not within the scope of the architecture, to decide and define the algorithms that will implement the desired functionalities described above. However, it is important for critical items of the architecture and related functions to consider the algorithmic approach (to provide hints for the implementation phase) as well as justify architecture choices which are depending on the algorithmic approach.

In this section we describe two basic issues that are demanding an algorithmic resolution. These are:

- **From Query Form to JP Finder**: This is the function that identifies whether the query involves answers from one or more JPs, which JPs will be asked and whether an international trip segment is necessary (if not given). The function is
also invoked when the User query is formed, in order to initiate the validation function as described earlier on, for parts of the query data

- **User Query Segmentation**: It is the function that formulates series of queries to JPs that will provide the partial answers giving the trip segment to examine. Many algorithmic solutions might be employed here. The decision should also take into account the international segments that are selected and are feasible within the criteria set of the user (according to the user query).

The paragraphs below go into deeper analysis of algorithmic considerations.

### From Query Form to JP Finder –

Let’s take an example of a user query:

**ORIGIN**: Italy, Florence, Piazza Stazione  
**DESTINATION**: Greece, Crete, Heraklion, 5th Ionias Street,  
**DEPARTURE DATE**: 25th July  
**GIVEN EXTRA JP SEGMENT**: Florence-Athens, airline ALITALIA, Departs at: 25th July, 10:00, Arrival at: 25th July, 13:00, local time

The JP Finder should find ATAF and ENOSIS as the candidate JPs to ask for validation of data.

An easy way for the JP Finder function could be based upon keeping Journey Planner metadata in all destination names that are covered. A simple search within these data could get the JP details for the JP which could be asked. However, this approach would need a tremendous maintenance effort, while the integration of any new Journey Planner would actually need a large amount of information, actually an updated dataset copy from the JP should be also available into the WISETRIP JP data. Though not an elegant solution, this approach provides easy validation of input data, whenever the user provides wrong input i.e. due to spelling errors. Therefore, the first finding is that street names cannot be kept in the JP data of WISETRIP. Update and maintenance will not be possible, and integration of JP would then require a huge amount of data and information.

A more efficient solution is required that will also secure all the following:

- Easy integration of a new JP
- Minimal data maintenance requirements
- A method to validate user input (location names)

Whatever, that solution would be, if the location details are not locally available into the WISETRIP dataset, there should be a way to ask the determined JP whether the input location data are valid. Therefore, the “Validate User Query Data” Function has been made necessary within the functional architecture analysis made. To identify candidate JPs we should utilize knowledge about a wider area domain that is covered by the JP, i.e. country level information, region level information or major cities level.

So, we could somehow represent the following information for the two JPs of the example:

- Italy, Florence, Metropolitan Area level → JP ATAF → buses, trams, railway
- Greece, All Cities, National Level → JP ENOSIS → ferries, buses, airlines.

The string representation could also include an indicative list of major locations at specific level. Such representation might point to more than one candidate JPs. In that case, the validating function will be utilized to find out those JPs that can provide answers for the specified query data. Of course it might be the case that more than one JPs can provide response for the same (or different segment). However, this is not a problem for the JP finder function which could possibly indicate more than one JPs. Anyway, the whole algorithm and data representation should be carefully designed in order to realize the JP finder function.
**User Query Segmentation** – When we think of User Query segmentation, we can identify two basic algorithmic approaches as described below:

1. **User-Specific Segment View:** The individual segment queries carry the specific user criteria (time, number of hops, duration) and preferences to the Journey Planner level. All queries are issued for each one extra JP/international segment. This approach produces more queries, engages more complex algorithms, need more JP data, has more risks in facing heterogeneous JP capabilities, but produces more accurate results.

2. **Broader Segment View:** The individual segment queries do not carry user criteria in detail and have looser time limits. This approach produces less queries, in a simpler way, produce more generic format queries (which can be accepted from almost all JPs) but produce a wider set of results that needs to be processed at a later stage in order to take into account the user specific preferences (as described within the user query data).

![Query Segmentation Diagram](image-url)

The figure above illustrates in a generic way the query segmentation phase. In this specific case we have two Extra JP / International Segments:

\[ X \rightarrow Y \text{ and } X \rightarrow M_1 \rightarrow Y \]

Each one has specific time requirements \(X(t)\) and \(Y(t)\) for arrival and departure. Queries to complete the route for \(A \rightarrow X\) and \(Y \rightarrow B\) segments, can be issued under the specific time requirements \(X(t)\) and \(Y(t)\). However, this might fail if the JPs do not support such time restrictions. Alternatively, the system could avoid timed specific queries and just ask for \(A \rightarrow X, Y \rightarrow B\) and \(X \rightarrow Y\) and then check how and which answers can be combined.

The table below describes in principle the two different algorithmic approaches.
3.1.3 WISETRIP Personalisation Functions

The personalisation functions of WISETRIP aim to detect the information that must be sent to the user according to his/her profile and the trips cycle criteria selected by the user, process alerting messages, produce notifications and define when and how to send a notification message. The personalisation functions interact with external information sources, user data and profile through a real-time decision mechanism, which processes all external and user data.

Personalisation functions are analysed in detail in the following sub-sections and are separated to those related to real-time decision mechanisms and those related to the management of personal trip life cycle information.

3.1.3.1 Real-Time Decision Functions

The WISETRIP Personalisation Functions include a set of Real-Time Decision Functions which will be available all through the duration of the trip. These functions receive real-time information on a permanent or on request basis and provide the interested users with critical information, depending on their preferences. These functions are, therefore, interactive functions that will be operating during the trip in different phases of the trip life cycle.

Different types of Real-Time Decision Functions are defined, depending on the phase of the trip cycle in which the function is going to be needed. In the following table a description of each one of the Real-Time Decision Functions defined in WISETRIP will be provided, as well as the type of data each function will require.

<table>
<thead>
<tr>
<th>Nr</th>
<th>Title &amp; Brief Description</th>
<th>Name of module to implement the function</th>
<th>Data Required / Other Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Select one trip from the list of available responses</td>
<td>The user interface passes a select trip, chosen by the user from the list of available trips found within the WISETRIP response data.</td>
<td>The main data used are the WISETRIP Response data. Function interacts with the user interface</td>
</tr>
<tr>
<td></td>
<td>Function</td>
<td>Description</td>
<td>Module</td>
</tr>
<tr>
<td>---</td>
<td>--------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>---------------------------------------------</td>
</tr>
<tr>
<td>2</td>
<td>Produce personal trip life cycle</td>
<td>This function produces personal trip data, based upon the trips selected by the user and according to the rules implied by the trip life cycle model data. Function produces the Personal Trip Data</td>
<td>Personal trip data generator</td>
</tr>
<tr>
<td>3</td>
<td>Arrange schedule of notifications</td>
<td>This function provides personal trip data with timestamps and orders for notifications, according to the rules implied by the trip life cycle model data and the personal profile of the user.</td>
<td>Personal trip scheduling</td>
</tr>
<tr>
<td>4</td>
<td>Monitor trip life cycle</td>
<td>This is a continuous function that runs over all personal data and checks whether and when conditions are mature for a specific action to be taken.</td>
<td>Personal trip monitor</td>
</tr>
<tr>
<td>5</td>
<td>Get real-time data</td>
<td>This function forms a query to a Journey Planner to request specific real-time information (i.e. traffic or incident data) Function interacts with the Journey Planner Interface. Function is invoked by the Personal Trip Monitor</td>
<td>Get Real-time JP info</td>
</tr>
<tr>
<td>6</td>
<td>Receive Alert</td>
<td>This is a continuous function that awaits for the receipt of alerts from Journey Planners or other systems Alert format is necessary to be defined.</td>
<td>Alert listener</td>
</tr>
<tr>
<td>7</td>
<td>Notify Traveler</td>
<td>The function constructs a notification ordered by one of the monitoring functions. The function uses Template data, personal profile data and trip data to form the notification. The notification is passed to the appropriate interface so that to be delivered via SMS, text mail or other to the user.</td>
<td>Notification Generator</td>
</tr>
<tr>
<td>8</td>
<td>Check Alert.</td>
<td>When an alert is received, the system through this function checks it to decide whether and to which personal trip it is relevant and affects. Function interacts to Device Gateway.</td>
<td>Alert Checker</td>
</tr>
<tr>
<td></td>
<td>Function</td>
<td>Module</td>
<td>Data Required</td>
</tr>
<tr>
<td>---</td>
<td>---------------------------</td>
<td>---------------------------------------------</td>
<td>---------------------------------------------------</td>
</tr>
<tr>
<td>9</td>
<td>Produce &amp; Send Alert</td>
<td>Alert Generator</td>
<td>Data required include Profile data and Alert</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>templates.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>Redesign trip.</td>
<td>Trip redesign</td>
<td>Necessary data include personal trip data and</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>profile data.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>Validate a register trip.</td>
<td>Trip Validity Checker</td>
<td>Data required include Personal Trip data</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>It interacts and invokes the Produce &amp; Send</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>alert function.</td>
</tr>
<tr>
<td>12</td>
<td>Booking &amp; Ticketing</td>
<td>External Booking Systems</td>
<td>Data required are the WISETRIP response</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>data.</td>
</tr>
<tr>
<td>13</td>
<td>Store query and define</td>
<td>Periodical Query Store</td>
<td>Query data and Profile data are necessary.</td>
</tr>
<tr>
<td></td>
<td>periodical update</td>
<td></td>
<td>Function interacts with the User Interface.</td>
</tr>
<tr>
<td>14</td>
<td>Personal Updating</td>
<td>Personal Update Monitor</td>
<td>Data required include Stored queries and Profile</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>data.</td>
</tr>
</tbody>
</table>
Occasionally some changes can occur during the trip life cycle time that make it difficult, or even impossible, to accomplish the trip plan in one or more of its trip segments. When these changes occur, the system (or the user itself) might have to redesign the trip.

The changes or unforeseen events can be of many different kinds. Depending on the grade of this changes or events the re-designing of the trip planning will be done either automatically by the system or interactively by user requests.

- If the changes occurred are minor changes - such as a small delay in one of the transport modes, a change on the platform in which the user will have to take a bus/train, etc. or any other change that is not affecting any of the other trip segments - the trip re-planning will be done automatically by the system and it will inform the user about the minor changes arranged.

- On the other hand, if an important unexpected event occurs that makes it impossible to accomplish one or more trip segments – for example, the bus/metro service has been stopped for any reason, a plane has been canceled or an important delay appears that makes it impossible to arrive on time to change the transport mode – part of the trip, or even the whole trip, will have to be redesigned. In this case, the system can re-plan the trip but it will need the user’s confirmation for the new planning. Besides, the user has to be able to manually re-plan the trip if he/she does not agree with the proposed re-planning.

In some cases the user might also want to re-plan the trip even when no unexpected changes occur in the pre-planned trip. This can happen, for example, when the user is
delayed at some point during or before the trip. In this case the user will have to be able to introduce the expected delay he/she is foreseeing to have and the point of the trip where the delay is going to applied.

Through the abovementioned functionality, the envisaged system is ready to process alerting, whenever available by the participating JPs. Moreover, as already identified within D.2.1 “Report on the Analysis of Requirements”, alerting is not supported by most JPs. Therefore, the functionality is enhanced with means that can either a. enable the user to produce an alert based on his own findings and knowledge (even during the trip process) or b. monitor a specific trip periodically to identify changes of schedule (at pre-trip phase).

Similar are the cases of a) regular updating on a specific trip and b) storing of requests for not already scheduled travel, which are also not supported by most JPs. In that case, the functionality along with the necessary bookkeeping is made at the WISETRIP level, rather than it is expected from the JP level.

3.1.3.2 Personal Trip Life Cycle Management Functions

The modulation of Life Cycle for a specific trip concerns firstly, the identification of segments as defined in the Trip Life Cycle Model and then the determination of the data that concerns each segment. According to the user profile this data is segregated depending on his interest. Then a timetable is shaped in which, all segments of the Trip Life Cycle are placed temporally, as well as the relevant information that should be forwarded to the user. The Trip Life Cycle Manager follows the progress of the trip and based on the timetable knows in which segment it is currently found and what information should be sent to the user.

Specifically, in the whole procedure the following functions can be distinguished:

1. Initial Trip Life Cycle formation:
   - **Select A Trip** (function 1): The user interacts via the user interface and forms the initial Trip Life Cycle by selecting from the list of available responses. The Trip Life Cycle is based on the time details of the trip (departure and arrival time).
   - **Personal Trip Production** (function 2): The formation of the initial Trip Life Cycle can also be based on rules that affect transportation with different means. For example, deadline for a ticket payment or deadline for ticket control before a flight - check in.

Therefore, a specific Trip Life Cycle can easily be shaped a priori (from the moment of ticket reservation) following the trip schedule and the Trip Life Cycle Model presented above (see page 12 Section 2.3.2).

2. Determination of the Data for each segment of the Trip Life Cycle:
   - **Arrange Schedule of Notifications** (function 3): In order to formulate the notification schedule the system combines data from the personal trip data, the trip life cycle model data and the user profile data.

For each segment of the Trip Life Cycle the system distinguishes the data that could be forwarded to the user (Trip Life Cycle Model Data). Taking into consideration the User Profile only the information of user’s interest are selected and associated with the partial segments of the specific Trip Life Cycle.

3. Trip Life Cycle Manager Modules:
   - **Personal Trip Monitor** (function 4): After the formation of a Trip Life Cycle a mechanism is required in order to decide correctly which information and when, will be pushed to the user. The module proposed is the Personal Trip Monitor which is a real time decision mechanism that follows-up the trip progress. At each point in time it knows in which segment it is currently found and decides, according to the shaped trip life cycle the required actions to be taken.
Get Real-Time data (function 5): In order to validate certain actions before committing the Personal Trip Monitor must interact with the Journey Planner Interface. A query is made to a Journey Planner to request specific real-time information (i.e. traffic or incident data). The main data required for this query is the initial trip data (location/time). In order to push updated information for the specific trip relevant to the user, a scheduler for checkpoints/messages is included. The module implementing the above function is the Get Real Time JP Data module.

Notification generation (function 7): In order to notify the user, a function which constructs a notification ordered by the Personal Trip Monitor is required. The function must use template data, personal profile data and trip data to form the notification. The notification is passed to the appropriate interface so that to be delivered via SMS, text mail or other to the user. The module used to encapsulate this function is the Notification Generator.

If we assume the users follow all transportation rules and the trip follows the initial schedule, the system that follows-up the trip can forecast at each moment (real-time) in which segment the trip is found with relative precision. If however for any reason the trip deviates from this schedule then the Trip Life Cycle that has been shaped is rendered invalid. Therefore, further implementation is required in order for the system to be aware of these changes, take them into consideration, and re-shape the Trip Life Cycle for the rest of the trip. However, before introducing the modules responsible for the re-design of the trip we should consider the modules that are required to generate the trip re-design.

4. Events Listening:

Alerts Listener (function 6): The Trip Life Cycle Manager can access external data sources and receive interesting information that might affect a trip progress. Whenever the system is aware of an extraordinary event (for example delay in the arrival of a route, weather alerts etc.) it examines if this event affects at this particular time any Trip Life Cycle that is in process and decides for further actions (push the information to the relevant users or even reshape the Trip Life Cycle). Therefore, the proposed module is an Alert Listener, a continuous function that waits for the receipt of alerts from Journey Planners or other systems.

5. Trip Life Cycle Manager Modules:

Check, Produce & Send Alert (function 8 and 9): Due to a number of reasons as mentioned above the system receives an alert. The system must then decide whether and to which personal trip this incoming alert is relevant and affects. This validation is done by combining information from the personal trip data, the user’s profile and the notifications data. To realize this function the Alert Checker module is proposed. Having checked the incoming alert the system must send the alert in the appropriate format to the user. The alert must be produced according to the profile data and the predefined alert templates. The module implementing this function is the Alert Generator.

Validate a register trip (function 11): At any point in time during the Trip Life Cycle a trip may be cancelled or changed. Therefore a mechanism that will continuously monitor a personal trip must be implemented. It must receive the personal trip data as input. This monitor should also be able to generate an alert in order to inform the user of such changes/cancellations. Therefore, the monitor must interact and invoke the Alert Generator module. The module proposed is the Trip Validity Checker module.

Trip Redesign (function 10): During a user’s trip there can be a drastic event which would have serious repercussions to the rest of the Trip Life Cycle. In order to compensate for this type of events the system must consider the redesigning of a trip.

Trip redesign may be initiated for three reasons:
An incoming alert, the Personal Trip Monitor initiates the redesign

- The Trip Validity Checker module identifies that the trip has changed
- The user himself initiates a trip redesign through the User Interface

The necessary data required by the redesign function is the Profile Data and the Personal Trip Data. In the first two scenarios depending on the incoming alert type, or the magnitude of the change in a trip a new trip is produced by the system according to the data provided. Furthermore, the user depending on his profile preferences might be asked again by the system for alternative trip solution. There is also the possibility the user is asked to choose among a list of new trips. In the third case scenario a user might initiate a redesign process for reasons outside the scope of his initial trip (e.g. met a friend whilst island hoping in Greece and requires a redesign in order to conform to his friend’s schedule.). The system must be able to produce the newly formed trip life cycle and cancel the previous one. The module proposed to encapsulate the above is the Trip Redesign module.

6. Ad-hoc Services:

- **Store Query** (function 13): The user has the option to store a selected trip via the user interface. The selected trip can be periodically updated based on rules defined by the user in his profile. For example, a user has stored a query of a particular trip (PATRA-GREECE to ANCONA-ITALY by boat with a specific company). The user may wish to be notified about new companies implementing this route. This function is implemented in the module Periodical Query Store

- **Personal Updating** (function 14): Requires input from profile and query data. By monitoring stored queries and profiles decisions are made about when a personal update must take place. If conditions are mature, it initiates a query process that finally leads to information sent to the user according to his profile.

  For example, a user has stored a query of a particular trip (PATRA-GREECE to ANCONA-ITALY by boat) however the company, responsible for this route has removed the particular route from its schedule. An alert will be issued notifying the user for this long-running change. The module responsible for the implementation of this function is the Personal Update Monitor

- **Booking & Ticketing** (function 12): This function falls within the family of personalised WISETRIP functions, though it is initiated through the core services. This is the function of taking the link to on-line booking systems, provided within the WISETRIP response data. More detailed description of the booking functionality is given in Section 3.2.1.2

3.1.4 WISETRIP Services Functions

WISETRIP Services layer aims to realise and put functions of Core and Personalisation Systems into action. Its functionality includes a set of calls necessary for pulling (on demand) or pushing (automatically) information to the end user.

More specific, the WISETRIP Services layer activates the following functions:

1. Form user query
2. Form extra JP / international query
3. Provide a predefined international segment
4. Filter and present journey planning responses
5. Select a trip and produce personal trip life cycle
6. Notify traveler
7. Produce and send alerts
8. Redesign a trip
9. Booking & Ticketing

10. Store query and user profile data

All the above functions have been presented in details in Tables 1 and 3.

### 3.1.5 Language Adaptation Function

The vision of WISETRIP as a common reference point (a “one-stop” office) for journey planners, with an integrated and unified interface over a network of participating Journey Planners of a specific coverage (usually urban, national, regional), engages the needs of an international market. Henceforth, multilingualism should be supported, which however cannot be guaranteed at all levels, as not all Journey Planners are having multilingual capabilities, neither can they have the same support to the same set of languages. We can distinguish the need for multiple languages at the query form level and at the response level.

<table>
<thead>
<tr>
<th>Nr</th>
<th>Title &amp; Brief Description</th>
<th>Name of module to implement the function (when applicable)</th>
<th>Data Required / Other Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Choose Language</td>
<td>The user selects a language of his/her preference. The language selected is given either at the query form phase, or it is also given when the user specifies his/her profile.</td>
<td>Function interacts with the user interface where the language choice is made.</td>
</tr>
<tr>
<td>2</td>
<td>Language query</td>
<td>When passing a query to a JP we notify which is the language of the data submitted and which is the preferred language of the response.</td>
<td>Functions interact with the JP interface where the language choice is informed to the JP and the JP responds either having the data in the preferred language or without any translation.</td>
</tr>
<tr>
<td>3</td>
<td>Translate Response</td>
<td>Having got the WISETRIP response, and knowing the preferred language at the query level, the system performs an automatic translation of the data. Symbolic representation of the answer will be utilised at the presentation level of the interface in order to minimize translation need. <strong>Module: Automatic Response Translator</strong></td>
<td>WISETRIP response data and multilingual translation data are necessary.</td>
</tr>
<tr>
<td>4</td>
<td>Translate Alert or Notification</td>
<td>This function translates each alert and notification into the preferred language. The alerts and notifications will contain symbolic representations that will not need translation. <strong>Module: Message Translator</strong></td>
<td>Notification and alert template data at multiple languages.</td>
</tr>
</tbody>
</table>
3.2 WISETRIp External Systems

3.2.1.1 Participating Journey Planners functionality

The basic service of WISETRIp is the search for the optimum routes and journey planning on a specific User Query for trip information. However WISETRIp will not replace any existing Journey Planner system, on the contrary it interacts with pre-existing Journey Planners to get the necessary information.

The WISETRIp Core System is connected to many JPs of variant types, which are serving different or overlapping areas, controlled by different transport operators or content providers and have different functionality (extensive or more restrictive). WISETRIp is not focusing on the particularity of the participating JPs, but to the activation of a single interface through which it will communicate with the participating JPs and search for the transportation routes that the end user prefers. The main services of this interface and their corresponding functions are listed in details in Section 3.4.1 of this report.

3.2.1.2 Booking & Payment functionality

Booking and payment functionality is among the key features for systems like WISETRIp, which can enhance the added value of the system and its market acceptance. However, a unified booking and payment mechanism would face significant challenges which most possibly could lead to dead ends, as commercial policies of the transport operators can be complex, technical limitation and diversities; there is no unified ticket.

A multi-modal combined booking system, that could combine multiple operators into a single ticket, enable co-operative policies (i.e. automatic discounts for special combinations) within an open environment, is a challenging project that is beyond the scope of WISETRIp, though it should remain among its long-term vision.

For this reason, the WISETRIp architecture foresees, booking and ticketing functions to take place within the external systems of operators, linked to the trip information that the user finally gets from WISETRIp. A link to the system could be two ways,

a. either a simple URL link to the booking page of the operator. The user should browse and fill all appropriate data in order to find and retrieve availability and price data, for the trip that is included in the WISETRIp response data.

b. or a link that will also be informed with the trip data for the segment presented in the WISETRIp response data, so that the page of the booking site can be already filled with the necessary trip details for the segment.

The Figure 7 below illustrates the envisaged functionality of WISETRIp, related to booking and on-line ticketing. Actually, the booking functions are performed by the external system. WISETRIp functions assure the link to the right operator’s service, with or without the trip co-ordinates that are already pre-selected by the user, included within the WISETRIp response data he has got from WISETRIp.

It is obvious that the Journey Planner data should be either informed about such booking site information (URLs or other data) or the Journey Planners themselves should convey this type of information within their segment responses.
Consider the above schema; the user has entered a query giving information about a trip he/she desires to take. The WISETRIP system responds with a list of solutions according to the user’s query. Each trip solution comprises of different segments, some of which may offer the option of on-line booking to the user. If this is the case, an “on-line booking” button will appear beside the partial trip details. The reason for only some of the segments presenting this option is that urban (local) bus tickets, taxis, or short train rides may not offer on-line booking capability. The user can then proceed to book the specific segment of his trip by clicking on the button. The WISETRIP system can pass the information (trip details i.e. date of departure/arrival, number of passengers, vehicles etc) to the on-line booking system. Therefore the user has the option to repetitively book the appropriate segments of a journey. The WISETRIP core system’s obligation would be to redirect the user to the appropriate booking engine depending on each segment independently.

Consider a real life example of a booking option from Helsinki, Finland to Santorini, Greece. The trip would comprise of 4 segments:

- A bus ride to Helsinki airport from place of residence
- A flight from Helsinki airport to Athens airport
- A metro ride from Athens airport to Piraeus port
- A boat trip from Piraeus to Santorini island.

In a probable scenario the booking button would appear on two of the segments of the user’s trip: the second and the fourth part of the trip. The on-line booking system of FORTH-CRS supports all domestic ferry transportation within Greece. The fourth segment of the trip mentioned above could therefore redirect the user to FORTH-CRS booking system, where he would have option of booking part of his trip.
3.2.1.3 Other data providers

In Section 2.2 of this report the participating services and source data layer of the WISETRIP architecture is described. This introduced the intention to utilize existing systems (such as journey planners or booking systems), as well as external information sources and services (such as time, user position, external alerts). This forms the essential data for the core functionality of the WISETRIP system.

One of the important aspects for the development of the WISETRIP system is the provision of additional external data from these sources in a consistent format which WISETRIP can then utilise. This data can be the information about ports, airports, train stations and even car park facilities and further detail of the condition of each of the transport node such as weather, tourist attractions, points of interest, retail opportunities, etc, all of which add value to the user response.

Table 5 below demonstrates various services that can be offered for different transport modes.

<table>
<thead>
<tr>
<th>Mode of transport</th>
<th>Data provided</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Routes</td>
</tr>
<tr>
<td>Car</td>
<td>X</td>
</tr>
<tr>
<td>Bus</td>
<td>X</td>
</tr>
<tr>
<td>Train</td>
<td>X</td>
</tr>
<tr>
<td>Flight</td>
<td>X</td>
</tr>
<tr>
<td>Bicycle</td>
<td>X</td>
</tr>
<tr>
<td>Walk</td>
<td>X</td>
</tr>
<tr>
<td>Taxi</td>
<td>X</td>
</tr>
<tr>
<td>DRT</td>
<td>X</td>
</tr>
</tbody>
</table>

From Table 5 it can be seen that different transport modes have different data that needs to be provided by JPs. For example, car travel does not need data about schedule or fares but the presence of car park availability, street map and time and furthermore CO₂ emission would be desirable from the WISETRIP system. Additionally, this information for car trip would likely be of most use to a user if it were obtainable from a mobile environment (during a journey). The above example demonstrates that a different data requirement is required to the means of travel preferences.

Moreover, some multi-modal journey planners include links to the web sites of individual transport service providers. For example the Finnish Journey.fi (Matka.fi) service can be linked to the website of Finnish railway transport provider VR. Through that link the users of the journey planner can review the price information of the train tickets and even make reservations and purchase tickets for that segment of the journey that consists of train transport. In most cases the pricing of the tickets is not unambiguous; the majority of transport providers have different price classes for the tickets depending on e.g. the timing of the ticket purchase or timing of the actual journey (discounts for customers buying their tickets in advance, night time extras, holiday season discounts etc.) duration of the validity period of the tickets (tickets for longer time periods/tickets that are valid for several trips might be cheaper than single tickets) or the personal features of the customer (discounts for children, students, pensioners, disabled persons, veterans, regular customers etc.).

Although in principle it would be possible to take into account the personal features of the user (through the information available from the user’s personal profile) while defining the ticket prices, most of the pricing principles are quite complex and
dependable on so many variables that it is very challenging to give accurate price information for multi-modal journeys consisting of several stages.

3.3 Distribution & Presentation Functionality

The fundamental idea behind WISETRIP is to offer services before and during a user’s journey starting from simple on-trip SMS (Short Message Service) services combined with pre-trip Internet information services. The objective is to develop the necessary interfacing mechanism so that the realised services can be distributed and presented to the end users through the use of fixed or mobile devices connected to the internet, from the initial search of a route and the time of user’s departure, until the arrival at his/her final destination.

The user interfaces of WISETRIP have to support and attract new users and meet the following desired qualities:

- Reliable service provision
- Prompt response
- Best availability
- Minimum amount of errors
- Cost efficiency

Use of the system should be easy, fast, reliable and reasonably priced. The value added services need to create a competitive advantage, as will those that reduce cost or save time for the individual user at some point in the service provision chain.

When it comes to the devices and technologies that can be used in the WISETRIP, an indicative list can be presented:

- Internet, web services and portal for the trip and journey related information provision and maintenance (including registration & personal profiling)
- Fixed and wireless communication networks such as GSM, GPRS and UMTS for supporting the information provision
- GIS (Geographic Information System) environment for geographical information and service references
- GSM, GPS (or the Galileo-based positioning system in the future)
- Mobile equipment for interacting with the service centre such as in-vehicle terminal, smart phone, PDA (Personal Digital Assistant), PC (Personal Computer), etc.
- Information and advice provision by mobile phone, messaging, SMS.

3.3.1 Fixed Web Functionality

3.3.1.1 General Web Functionality

The WISETRIP user interface refers to the graphical and textual information the system presents to the user, and the control sequences the user employs to control the program. The Graphical Users Interface (GUI) of WISETRIP will offer graphical icons, and visual indicators to fully represent the information and actions available to the user. The actions are performed through direct manipulation of the graphical elements. The components of the WISETRIP GUI will be:

- **the Pointer**, which is a graphical image in the shape of a gloved hand with outstretched index finger that indicates the location of a pointing device, and can be used to select and move objects or commands on the screen.

- **Menus**, which will allow the user to execute commands by selecting (with the mouse, keyboards or other pointing devices) from a list of choices. The menu bar is displayed horizontally across the top of the screen. The Menu bar will be composed of different buttons such as:

  Home, Plan a Journey, Live Travel, Log-in/Register.
A pull-down menu is associated with each button of the menu bar. When a user clicks on a menu option the pull-down menu will appear. The pull-down menus associated with the menu bar are:

- **Home:** FAQ | About us | Contact us | Sitemap | Related sites | Terms & conditions | Privacy policy | Data providers | Accessibility
- **Plan a Journey:** Door-to-door journey planner | Find a train | Find a flight | Find a coach | Find a bus
- **Log in/register:** Existing user | Register | Why should I register
- **Live Travel:** Live Travel News | Departure boards

- **Icons:** associated to each pull-down menu option

English will be used as the homepage language. There it the need for multiple languages at the following levels:

- Query Form Level
- Response Level

Flag-icons will be displayed on the top-right corner of the page according to the chosen language.

The tables below summarize and show potential examples of desirable graphical user interface features and behaviour. Particularly, the examples refer to the following use cases:

- submitting a user query to WISETRIP
- waiting for WISETRIP query results
- presentation of query results
- logging in / registering into WISETRIP
**MENU OPTION: Plan a Journey**

**Pull-down Menu: DOOR-TO-DOOR JOURNEY PLANNER**

<table>
<thead>
<tr>
<th>Primary Actor</th>
<th>User</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>The user wants a complete journey from an origin to a destination by joined-up public transport and formulates a query on the WISETRIP system</td>
</tr>
<tr>
<td>Triggers</td>
<td>Any journey from one place to another that requires a plan</td>
</tr>
<tr>
<td>Pre-conditions</td>
<td>The user must be connected. The user must know the O/D places as well as the departure time (and return if needed)</td>
</tr>
<tr>
<td>GUI</td>
<td><strong>- A Window</strong> (in a different color from the webpage) including all the functions and elements indicated below</td>
</tr>
<tr>
<td>Functions</td>
<td><strong>Indicate Origin (From) and Destination (To)</strong></td>
</tr>
<tr>
<td>GUI elements and position</td>
<td><strong>- a single line rectangular text box</strong> displayed on top of the window with a border that separates it from the rest of the interface to input the O/D address</td>
</tr>
<tr>
<td></td>
<td><strong>- a button “Find on Map”</strong> (placed next to the text box) to visualize the origin/destination place on a map</td>
</tr>
<tr>
<td></td>
<td><strong>- the option “unsure of spelling”</strong> (placed next to the text box) which can be ticked in case of uncertainty on the spelling of the destination</td>
</tr>
<tr>
<td></td>
<td><strong>- Different options</strong> placed under the text box that can be ticked, according to the information typed in the text box: <strong>Address/street/postcode</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Country/City/Area</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Facility/attraction</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Station/airport</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Stops (eg. Bus, Tube, Tram, etc.)</strong></td>
</tr>
<tr>
<td>User action</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>-------------</td>
</tr>
<tr>
<td>1) The user types the O/D address on the text box and clicks on the button “Find on Map” to visualize its position on a map.</td>
<td></td>
</tr>
<tr>
<td>2) The user may tick the option unsure of spelling if this is the case.</td>
<td></td>
</tr>
<tr>
<td>3) The user can click one of the other options according to the type of address inserted in the text box.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>GUI elements and position</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Five Spinners</strong> (one next to the other) oriented vertically with an adjoining text box displaying:</td>
<td></td>
</tr>
<tr>
<td>- the first, the day of the month from 1 to 31</td>
<td></td>
</tr>
<tr>
<td>- the second, the month of the year</td>
<td></td>
</tr>
<tr>
<td>- the third, the option “Leaving at” or “Arriving by”</td>
<td></td>
</tr>
<tr>
<td>- the fourth, the hour from 00 to 23</td>
<td></td>
</tr>
<tr>
<td>- the fifth, the minutes from 1 to 60</td>
<td></td>
</tr>
<tr>
<td><strong>A calendar icon</strong></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>User action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>The user indicates the departure/return day and time by either clicking on an up or down arrow on the spinners. The day and time selected with the spinner will be displayed in the text box adjoined to the spinner.</td>
<td></td>
</tr>
<tr>
<td>The user can plan journeys for the current calendar month and the next two months.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Proceed with the user query</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Next button</strong>, placed at the bottom of the window, to be clicked in order to proceed with the query</td>
<td></td>
</tr>
<tr>
<td>Amend information</td>
<td>User action</td>
</tr>
<tr>
<td>-------------------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| GUI elements and position | - **Advanced Options button** to amend some options for the journey  
- **Different Windows** for each option that can be amended. The Windows will be displayed one after the other:  

1) **Window for the choice of the transport mode**: it displays a list of transport mode (train, coach/bus, underground/metro, ferry, plan, tram/light metro) which can be ticked by the user. User can tick one, more than one or even all types of transport and obviously his/her choices will affect the number of journeys found.  

2) **Window for the Public Transport Journey Details**: contains the following sections:  
- Changes  
- Walking speed (average, slow, fast)  
- Journey options  
  - Trip Duration (exactly, approximately, greater than, less than)  
  - Number of hops/limit number of hops  
  - Special Requirements i.e. E&D, vegetarian, green routes  
  - Cost Limit  
  - Number of Persons  
- a **Clear Page button** to cancel the user query |
### User action

The user can integrate the query by inserting the following details:

- Tick the transport mode he is most willing to use (1st window)
- Limit the n° of changes (2nd window)
- Indicate walking speed and time he is willing to walk for the journey (2nd window)
- Indicate the station/airport he wants to travel via (2nd window)
- Specify car details (3rd window)

### Waiting for query results (2nd screen to appear)

<table>
<thead>
<tr>
<th>Primary Actor</th>
<th>WISETRIP System</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>The WISETRIP System processes the user query and calculates the needed algorithm in order to find the best transport solution</td>
</tr>
<tr>
<td>Triggers</td>
<td>A user query must have been submitted</td>
</tr>
<tr>
<td>Pre-conditions</td>
<td></td>
</tr>
</tbody>
</table>
| GUI           | **An indeterminate progress bar**, placed in the middle of the page, which uses motion or some other indicator to show that progress is taking place and is followed by a sentence “Thank you for waiting while WISETRIP searches for your journey options”.

### Presentation of query results (3rd screen to appear)

<table>
<thead>
<tr>
<th>Primary Actor</th>
<th>WISETRIP System</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>The WISETRIP System provides a response which is the result of the search. The user selects (tick) the most appropriate solution.</td>
</tr>
<tr>
<td>Triggers</td>
<td>The search for the journey has been completed according to the options indicated by the user.</td>
</tr>
<tr>
<td>Pre-conditions</td>
<td>A user query must have been forwarded to the system.</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------------------------------------------</td>
</tr>
<tr>
<td><strong>GUI</strong></td>
<td><strong>Buttons:</strong></td>
</tr>
<tr>
<td></td>
<td>Details</td>
</tr>
<tr>
<td></td>
<td>Summary</td>
</tr>
<tr>
<td></td>
<td>Maps</td>
</tr>
<tr>
<td></td>
<td>Ticket costs</td>
</tr>
<tr>
<td></td>
<td>Modify Journeys/Amend</td>
</tr>
<tr>
<td></td>
<td>New Search</td>
</tr>
<tr>
<td></td>
<td>Print</td>
</tr>
<tr>
<td></td>
<td>Save / select journey</td>
</tr>
<tr>
<td></td>
<td>Booking</td>
</tr>
<tr>
<td></td>
<td>User Notification</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Functions</th>
<th><strong>GUI elements</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Show the result of the query</strong></td>
<td><strong>Details button</strong></td>
</tr>
<tr>
<td>GUI elements</td>
<td>Icons for each transport mode to be used in the selected journey</td>
</tr>
<tr>
<td>User Action</td>
<td>The user can select (tick) the option he prefers and can see the details of the chosen journey, including any connections to make, stations names, interchange times and/or driving instructions if the car journey has been selected.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Functions</th>
<th><strong>GUI elements</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Summary button</strong></td>
<td><strong>Summary button</strong></td>
</tr>
<tr>
<td>GUI elements</td>
<td>Table indicating for each option the modes of transport, the number of changes, the departure and arrival time</td>
</tr>
<tr>
<td>User Action</td>
<td>The user can see the route maps</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Functions</th>
<th><strong>GUI elements</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Maps button</strong></td>
<td>Graphic visualization of the journey on a map</td>
</tr>
<tr>
<td>GUI elements</td>
<td>The user can see the route maps</td>
</tr>
<tr>
<td>User Action</td>
<td>The user can see, if it is available, the overall ticket cost of the journey or the cost for a specific trip segment</td>
</tr>
</tbody>
</table>
### Amend the journey

**GUI elements**: Modify Journeys/Amend button

**User Action**

You can find the main part of your journey and then extend the plan. If you would like to allow more time at the places where you have to change transport, you can adjust your journey.

**GUI elements**: New Search button

**User Action**

### Print the selected journey

**GUI elements**: Print button

**User Action**

The user can click on the button to print the results of the journey research.

### Save the selected journey

**GUI elements**: Save / select Journey button

**User Action**

If the user is registered he can save the chosen option.

---

**MENU OPTION-LOG IN/REGISTER**

**Pull down menu: Existing user**

<table>
<thead>
<tr>
<th>Log in</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Primary Actor</strong></td>
</tr>
<tr>
<td><strong>Description</strong></td>
</tr>
<tr>
<td><strong>Triggers</strong></td>
</tr>
<tr>
<td><strong>Pre-conditions</strong></td>
</tr>
<tr>
<td><strong>GUI</strong></td>
</tr>
<tr>
<td>Functions</td>
</tr>
<tr>
<td>-----------</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>

**Pull down menu: Register**

<table>
<thead>
<tr>
<th>Register</th>
</tr>
</thead>
<tbody>
<tr>
<td>Primary Actor</td>
</tr>
<tr>
<td>Description</td>
</tr>
<tr>
<td>Triggers</td>
</tr>
<tr>
<td>Pre-conditions</td>
</tr>
<tr>
<td>GUI</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Functions</td>
</tr>
<tr>
<td>-----------</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>
3.3.1.2 Personalised Web Functionality

Personalised web functionality incorporates the user profiling editing session as well as a private section where the user can indicate his journey specific preferences.

The relevant GUI will follow the same components, features and behavior as described in the previous section (3.3.1.1 General Web Functionality). A potential example related to personalised functionality is presented below:

Pull down menu: My Profile

<table>
<thead>
<tr>
<th>My Profile</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Primary Actor</strong></td>
<td>Users</td>
</tr>
<tr>
<td><strong>Description</strong></td>
<td>The user is connected to the WISETRIP platform and has logged in to the system.</td>
</tr>
<tr>
<td><strong>Triggers</strong></td>
<td>Need to alter current profile add user preferences.</td>
</tr>
<tr>
<td><strong>Pre-conditions</strong></td>
<td>Availability of a PC and internet connection.</td>
</tr>
<tr>
<td><strong>GUI</strong></td>
<td><strong>My Profile Button</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Functions</th>
<th>Change Details</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>GUI elements</strong></td>
<td>Change Details Button.</td>
</tr>
<tr>
<td>Text Boxes for entering:</td>
<td></td>
</tr>
<tr>
<td>- existing password</td>
<td></td>
</tr>
<tr>
<td>- new password</td>
<td></td>
</tr>
<tr>
<td>- password confirmation</td>
<td></td>
</tr>
<tr>
<td>- email and/or mobile phone number</td>
<td></td>
</tr>
<tr>
<td>Check Boxes for entering:</td>
<td></td>
</tr>
<tr>
<td>- method of contact email/SMS</td>
<td></td>
</tr>
<tr>
<td>- method of receiving notifications Push/Pull/both</td>
<td></td>
</tr>
</tbody>
</table>

| User action | The user types his new password to alter his existing password and alters other personal information (email and/or mobile phone number which will be used by the system to provide information on a selected trip) |
### My Preferences GUI elements
- **My Preferences Button** to define preference when searching for a journey
- **Different Tabs** for distinct set of preferences:
  1) **Tab for the choice of the transport mode**: it displays a list of transport modes (train, coach/bus, underground/metro, ferry, plan, tram/light metro) which can be ticked by the user. Un-ticking these options may limit the number of journeys found.
  2) **Tab for user requirements choice**

### User action
The user edits his travel preferences by using select and checkboxes in the provided tabs.

### My Journeys GUI elements
- **My Journeys Button** to view registered journeys within WISETRIP.
- **Select Journey Button**
  - displays a list of notifications corresponding to the journey (date, time, type of notification) each notification is highlighted according to importance.
- **Delete Journey button** to remove the selected journey.

### User action
The user views a list of journey he has registered for WISETRIP support. The user selects a journey register in WISETRIP. The user views a list of notifications he/she will receive for the duration of the journey. The user deletes a registered journey.
3.3.2 Mobile Functions

The traveler/user of WISETRIP has access to this information through multiple interfaces such as a fixed web interface (web portal), a mobile interface (mobile device or PDA & mobile web-browser), and a messaging interface (mobile device or PDA & SMS). This section deals with the mobile interfaces and related architecture. The user interface architecture for the mobile users will be based on the existing standards and made available at the end of the project as (mostly) publicly available open source code.

So WISETRIP aims, in addition to fixed Internet information services, to provide the users with mobile, personalised and location integrated journey planner services. User friendliness is imperative for any system and functions to be developed (mobile or fixed) where user interface exists. The system should be usable by all. Thus WISETRIP user interfaces should also take into account the needs of special user categories (such as elderly and disabled people with physical limitations). It has also been stated that tourists and travelers should be provided with information in their own native language. Thus the objective of the WISETRIP is to develop ‘a consistent mobile user interface’. The WISETRIP data and services should also take into account the user profiles and preferences so that the data to be provided is customised to the needs of an individual user.

The WISETRIP system aims to create and offer a so-called ‘one stop-shop’ for the user of journey planning and related services. To achieve this, WISETRIP will be connected to a wide variety of JP databases and combine their data, based upon user requests and preferences (user profile). The combined data are geo-referenced in relation to the user's location, supporting when the user is on the move and uses mobile functions of WISETRIP. Most important of all is the fact that the system is 'intelligent'. In other words, it searches for appropriate data, according to the user preferences. If needed (and accepted by the user) it will offer the end user alternatives and suggestions related to his/her trip-cycle. The WISETRIP mobile functions should also take into account the user’s preferences and special needs (i.e. language, font size, etc.) and support integrated services where possible (booking, ticket purchase, etc.).

Definition and specification of the interface architecture for the mobile users of WISETRIP is very important since the end-users will use the WISETRIP system very often on-the-move using their mobile devices. Generally taken the user interfaces of WISETRIP should take into account or at least be aware of all the existing and related European level and national system architectures, technologies, regional and local data sources and interconnectivity protocols that are related to the WISETRIP service provision. Previous projects, for instance DOM (Der Oriente Mensch), ISCOM, Image etc. have stated that different mobile devices may need unique user interface solutions, but on the other hand, the same look and feel of the user interface for different services is of the utmost importance. Based on previous studies there are two main facts that strongly support the future chances of commercial success for WISETRIP. Firstly, personal mobile devices will most likely in the future become the primary access gate to Internet. Signs of this can be seen already now. Secondly, the value added services have increasingly grow in importance. This means that the current market situation will radically change by the advent of unprecedented opportunities for smaller companies specialised in the provision of on-the-move high quality contents and services.

The WISETRIP user interface architecture for mobile users will consist (as with any architecture) of three layers:

- device gateway (the presentation layer)
- the application layer/business logic layer
- the data layer/persistence and integration layer.

The presentation layer will be responsible for the interaction between the WISETRIP system and the mobile user. The user’s access to the WISETRIP server will thus be
through a mobile device (or a web browser). The user should not be bothered with a variety of different interfaces, meaning that the mobile WISETRIP interface should be standard regardless of the variety of databases and systems it accesses. In this way, a common interface will be implemented.

The services can be delivered to the user via the server side enabler technologies. The trip data be delivered to the user either in HTML format (in the case of a mobile www browser), in WML format (in the case of a wap-enabled phone) or as a SMS message. The mobile (client) hardware will define the software to run on it.

One of the mobile phone operating systems is Symbian (S60 & S80). The Symbian OS is compatible with the Personal Java and JavaPhone specifications. It enables to develop fully Java compatible applications. Moreover, the Java2ME implementation is being adopted by more and more mobile phone manufacturers and it will be implemented on most 3G telecom devices because SymbianOS is designed to run on devices combining PDA and mobile phone characteristics. In this way, the developed code will operate and run on the next generation mobile devices. However, also Palm OS, Windows CE can be used in WISETRIP. Windows CE has also proven to an effective operating system that makes it possible to develop high quality mobile applications. All the operating systems can be treated equally.

As far as the application layer is concerned, the various modules can be developed using component-based development technology. This is due to the fact that the ideas and guidelines of RUP (Rational Unified Process) will be followed. RUP is an iterative process and in order to achieve such implementation RUP suggests using a component development approach. The available technologies that can be used for component development are the EJBs (SUN), CORBA (OMG), DCOM (Microsoft). The full relation of the EJBs with the Java language suggests the former to be the adopted component development technology for the WISETRIP architecture.

The data layer of the system comprises of the databases and the functions needed to access them. As a matter of fact the data layer will not be in the mobile devices, instead if that the data can be found either on the WISETRIP server or in the journey planners integrated to WISETRIP. The application layers will be communicating with the data layer in order to receive data from the various content and services providers. The various databases will be connected to the WISETRIP system using TCP/IP. The received data format will be in XML. Therefore, a common XML specification has to be developed. The received data in XML format will be transformed in the proper format for processing using XSL or API and then delivered to the user’s device.

3.3.2.1 General Mobile Functionality

WISETRIP will design and develop a suitable mobile user interface mechanism and functions that interact with the user in a consistent way across various types of mobile devices. The interface design should allow sufficient customisation and adaptation of the user interface by the end-user. This kind of approach is suitable for new generation hand-held devices, as it will permit appropriate flexibility in the choice of media for presentation to and from the user (text, graphics, keypad, voice, etc.).

With the existing mobile devices it is already possible to use GIS systems and to present geographical data. Thus WISETRIP could, if so decided, also offer graphical data (maps, etc.) related services and functions to the mobile users. Once made available they can then be presented by using the mobile user interface according to the individual requirements. These services can be provided by GPRS- and UMTS- based hand-held devices. Generally taken the development of the society ‘on the move’ and emerging technical opportunities provided by new mobile equipment, new communication standards and wireless protocols, etc. enables new interoperable solutions such as WISETRIP.
### 3.3.2.2 Personalised Mobile Functionality

As mentioned above the WISETRIP data and services should also take into account the user profiles and preferences so that the data to be provided is customised to the needs of an individual user. Thus each individual user should be able to present his personal requirements and needs that will be taken into account not when planning but also during the trip. It has also been stated that the end-users (also mobile) - tourists and travellers - should be provided with information in their own native language. Thus the objective of WISETRIP is to develop ‘a consistent mobile user interface’. The WISETRIP Personalisation System performs journey planning functions and procedures according to the user’s profile and trip life cycle criteria selected by the user. The end user receives (push or pull) the needed information based on the ‘rules’ of the personalisation system.

It has been defined in the section 2.3.2 that the Real-Time Decision Mechanism constitutes an event listener application that receives real-time information on a permanent (or on per request) basis in order to decide what information will be ‘pushed’ to the users (mostly ‘on-the-move’). The Trip-Life Cycle Model takes into account the pre-registered trips selected by the users, and specifically the different periods of a journey (trip – specific life data) that constitute the Trip Life Cycle Model (Pre-trip phase, En-route phase). It defines the rules on which the trip analysis will be made into phases, so that a specific trip of a user is translated to Trip Cycle data. The decision mechanism includes a scheduler of checkpoints/messages for a specific trip that ‘push’ updated information relevant to each user.

The User Profiling is a provision mechanism for the necessary personal information for a selected trip that takes into account the configuration of a user's profile that will determine the activation of the appropriate Device Adaptor in order to provide access to the system, and allow the Trip Life Cycle Manager to specify this information that corresponds to the pre-defined user preferences. Each journey is scheduled according to the User Profile and the criteria of Trip Life Cycle selected by the user. Every trip is registered into a user profile and a specific trip life cycle for a specific user, based on the applied trip life cycle model, is created.

The final journey plan is produced by the Core System and handed to the Services of the WISETRIP Platform for delivery (or to the Personalisation System for further processing). Thus the services of the WISETRIP Platform communicate with the different end user devices through an intermediate layer which is the ‘device gateway’ that takes care for the open and harmonised interconnection to various devices.

### 3.3.3 Functional Implementation Considerations

The Functional Implementation is a critical phase in the WISETRIP project work since in this phase all the communication technologies and tools, technical infrastructure, devices, software modules of the WISETRIP system, etc. will become reality and are taken into (pilot) use. The functional implementation builds and creates the functionalities that have been planned and developed by the project.

The WISETRIP design is based on the status and knowledge of the existing journey planners, user needs and requirements, the WISETRIP consortium partners’ business and research objectives and initiatives and is influenced by the state-of-the-art assessment. The work done so far has identified trends, possibilities and experiences in functional and technology solutions related to the WISETRIP system. The WISETRIP system architecture will take these trends, experiences and other experiences into account in the implementation phase. To make all this happen a proper implementation plan is needed. An important issue that should be considered in the context of functional implementation is the validation of the input given by the user. In those cases where the same origin/destination location is covered by several JPs and the input is verified by more than one journey planner two questions particularly should be taken into consideration:
1. what is the operations model when the input qualifies as such for the JP(a) but the JP(b) requires more accurate specification? and

2. there will most probably exist situations where the same input refers to different locations depending on the journey planner; how will the incorrect options be screened out?

Core functionality issues relating to functional implementation include some consideration about the co-ordinate systems. Since it is likely that different local and national JPs have different co-ordinate systems (and some JPs probably are not even connected to any co-ordinate system at all) some common system should be chosen. One option would be to transform the geographic information to co-ordinates, use that co-ordinate information as a search parameter in JPs and, when needed and when needed and required by the JP, to geo-code the location information to co-ordinates. Google Maps could possibly be used when testing this functionality in demo use. In any event, if the co-ordinates are used (which is probably the case), transformations between different co-ordinate systems will be necessary to support the local and national systems.

3.4 Software Communications Architecture

Within the following sections, the intention is to describe the fundamental communication paths among different modules of the architecture, either functional or data modules. Description of communications architecture includes the following:

- Identification of interacting components of the architecture
- Means of Communications (i.e. TCP/IP, UDP, SMS)
- Messaging Format Technology (i.e. SQL, XML, HTML)

It is not within the intention of the document at this level to define exactly messaging formats and communication protocols among processes and functions. The basic interacting components within WISETRIP architecture are:

- WISETRIP and Journey Planners
- Functional Components and Database
- User Interface functions and Devices
- WISETRIP and other External Systems

3.4.1 Interfaces to Participating Journey Planners

As described earlier, there are several functions, either in the Core System or in the Personalisation System, which interact with a JP. The mechanism that assures the necessary communication function with each JP is the Journey Planner Interface. The data from JPs which can be incorporated for WISETRIP service provision includes: routes, schedules and fares for bus/train/flight; facilities (i.e. car park, shops, point of interest, hotels) at the bus/train stations or at destinations; routes for car and bicycle; cities and street map. A detailed breakdown of this database can be seen in Section 3.2.1.3 on other data providers (Table 5).

The Journey Planner Interface should be based on Web Services technology. XML based Web Services will provide a link through the Internet to the travel planning servers of existing JP systems. The WISETRIP consortium will specify the main services it offers and publish the client side code for them using Web Services Description Language (WSDL).

Each Journey Planner can then develop the server side code to complete the interface and become a participating Journey Planner. This ensures there is no limit on the number of participating JPs and allows entry of new JP’s to become WISETRIP participants.
Comment: Although this outlook is justified, careful consideration should be made as WISETRIP should remain a motive enough for the owners of journey planners to start developing specific codes. The final result should be carefully designed so that the expected work load for candidate participating JPs should not prohibit extension of WISETRIP. Furthermore, significant changes to the data definitions or functionality of existing participating JPs are only likely to be recognised and incorporated by WISETRIP if the JP developers also have responsibility for updating the server side code of the API.

It might be the case that not all types of service interfaces are available by a specific JP. For this reason we should distinguish, interfaces into “Mandatory” and “Optional” ones.

Main services and their corresponding functions are listed below:

**Validate Query:** Here, through this service interface, the WISETRIP system passes to participating JP main elements of the query for validation. Location names will be validated, as well as language availability. Positive answers confirm that data are validated and language preference is accepted. Otherwise, negative responses or alternative suggestions (for location names when spelling errors might have been identified or for available languages).

**JP partial query:** When the language barrier can be addressed, WISETRIP will be able to interface with the participating JPs that should act as an ordinary journey planner, delivering information to WISETRIP for partial segment request, so that WISETRIP could finally provide service with extended cross border door-to-door information. Information such as duration of a trip or a journey can be itinerated to be given to the end users. Furthermore, departure and arrival time for a multi-modal transport service and a booking facility flag should be provided to give the end users choices for pursuing a trip. However, the D.2.1 Section 3 report noted that fare information provided by JPs is very limited, because this information is only provided for specific transportation services.

**Get real-time data:** Through this service, the JP will provide real-time incident (or any kind) information for a given segment. When a user has already booked a trip or a journey, real-time information – dynamic data – about transport problems (such as traffic accident, train or flight delay, etc.) becomes very important. This is especially true when the user is in a long trip/journey where such an interruption may cause other alternatives of trip/journey to be considered or even a stay in a hotel. However, it should be noted that a regular information update might not be available since many of the JPs are not providing such services as reported in D.2.1 Section 4. Another aspect that has to be carefully addressed by WISETRIP is the segment from point X to point Y as described in Section 3.1.1 of this D.2.2 report. Thus a traveler within his/her trip/journey from X to Y does not need real-time information in segment A to X nor Y to B.

**Get alerts:** Through this function, WISETRIP functions can receive alerting information from participating JPs. The result from the WISETRIP JPs survey as reported in D.2.1 and D.2.1a shows that the current available JPs are generally not providing real-time alerts to travelers, however this feature is what many JPs envisage for the near future. Acknowledging the growing number of mobile internet technology user, JPs services have to provide personalized updated information to anticipate global travelers, thus giving confidence for travelers to use their personalized JPs wherever they are. However, as, again, noted in D.2.1 that alerting function on JPs is not always supported. Anyway,

---

2 For WISETRIP to provide international door-to-door multi-modal journey planning services, multi-lingual interfaces will become very important. However, as reported in D.2.1 Section 4 regarding system constraints and limitations, not all journey planners are multi-lingual, therefore multi-lingual adaptation mechanism should be employed. Dynamic lingual adaptation is a key functional requirement for the architecture. Since many countries used English as the second language, it would be easier for WISETRIP to provide English as the second language rather then incorporating other languages. Neighboring countries languages would also be beneficial for meeting the regional market. For example, in Belgium, three different languages are set to be the national languages because of the geographically adjacent nations.
WISETRIP interface with JPs, could get –when, if available- alerting information from participating JPs through such interface function.

Initial register: Through this service, an initial registration is made of the participating JP into the WISETRIP system. Initial registration provides necessary data that will be maintained in the Journey Planner Metadata section (is described in later Section 4.1.1)

Confirm (segment) for the validating: Once the users are registered, the JPs will respond to validate the requested trip/journey information. The JP confirms that the trip is valid. Section 3.1.1 has explained detail of a journey segment and provides some analytical approach to address the problem.

Get Map Image (segment): This information will help the users to have a geographic sense of their trip/journey. The JP could provide map image for the segment in order to assist the user in having a visual representation of the route. If the service is not available or possible, WISETRIP can consider using a world-wide map service (like Google Maps) to get consistent map images covering the segment and the whole journey.

Get Location Map (x,y): This information will help the users to explore the potential destinations near a given location (usually the current location of a mobile user): point of interest, nearest terminal / station.

Push/pull (mobile personalization system): As described in Section 3.1.3.1, the push and pull service, is providing information to users/travelers, either by their request or by the JPs initiatives to leave interactive options for further decision from the users/travelers.

3.4.2 Interfaces to devices

Regarding the interfaces to devices, Section 2.3.5 has already described that the WISETRIP services must be accessible through different user devices and the passive system that implements these services provides access methods through a common API in order for the system to be constantly accessible. A middle application module (Device Gateway) is needed to adapt the communication between a device and the WISETRIP system. A Device Gateway can be an interface to SMS messaging server or to a mail application that takes over to deliver mail notification to users when it is necessary.

WISETRIP will design suitable interfaces for handling interaction with the user in a way that is consistent across various types of end devices, and more particularly PCs, PDAs, GSM, GPRS and UMTS mobile phone devices. This will entail a task-oriented approach to the interface design, leaving the various device-dependent I/O options open. This also permits sufficient customisation (to be decided) and adaptation of the user interface by any new user who wishes to plug into the WISETRIP system. Such a design is also particularly suited to new generation hand-held devices, as it will permit appropriate flexibility in the choice of media for presentation to and from the user.

When planning the interfaces to different end user devices it is important to take into account at least of the following issues:

- existing European level and national system architectures (that might affect WISETRIP)
- chosen end user technologies and their capabilities
- international, national, regional and local data sources (Journey Planners)
- interconnectivity protocols that enable the access to the regional and local data sources via the WISETRIP core system.

To be able to create proper interfaces, the communication links and technologies need to be defined. It has already been presented earlier that the WISETRIP users will use GSM, GPRS and UMTS network-based communications. The fundamental idea behind WISETRIP is to offer services on all level solutions from simple on-trip SMS services to
www-browser services offered by the modern mobile devices. The interfaces, whatever
the device to be used, should have the same kind of a WISETRIP approach so that the
users are able to use the WISETRIP system easily with different devices. This means
mark-up languages (xml, html, wml, chtml, etc.) need to be acknowledged. Also, all the
related existing and emerging standards and "de facto's" need to be acknowledged
(Bluetooth SIG, Cell Broadcast Forum, GENELEC, ETSI, IETF, ISO, LIF, OGC, W3C,
WapForum, etc.).

Basic requirements and objectives that can be placed for the WISETRIP interfaces are:

- WISETRIP needs to provide a consistent user interface based on the user
  requirements gathered, plans produced by the WISETRIP consortium and relevant
  standards and guidelines
- WISETRIP shall provide clear, legible, understandable information for the end
  users.
- WISETRIP will provide information using text (SMS), graphical representation
  (including icons) and/or combination of these two. Graphical form may include the
  use of maps
- WISETRIP will provide information in the language the user has selected in his/her
  profile or native language at the output location, if for some reason the user’s
  preferred language or the default language (English) is not available.
- WISETRIP will use standardised icons, when available,
- WISETRIP will take into account all the existing standards and guidelines relating
  to HMI (including visual display).

The WISETRIP Core System needs to be able to interface with any positioning technology
used by the end user device, whether it is network based (cell-id) or handset-based
(GPS).

When it comes to the mobile device interfaces the following interfaces and protocols can
be listed:

- Interface between WWW Server and PDA/Smartphone WWW client - (x)HTML
- Interface between WWW Server and WAP client – WML
- Interface between SMS Gateway and SMS Client – SMS

3.4.3 Database Communications

Database communications is necessary in order to achieve the required interaction and
functional performance of retrieving data from and storing data into the local database
entities of WISETRIP. Database communications will be either based on native SQL
statement, ODBC or XML queries over TCP/IP.

3.4.4 Communication interface to other external systems

All communication interfaces are crucial for the WISETRIP system. This also includes the
interfaces to external data sources or other functioning systems (additionally to
participating journey planners, device systems and WISETRIP database). These sources
could be both private and public and the data from them needs to be integrated and
processed so that they can be used in WISETRIP. These external databases or systems
will be connected to the WISETRIP system by TCP/IP. The data format to be used will be
in XML. Therefore, a unified XML specification needs to be developed. If/where needed
the received data in XML format will be transformed in the proper format for processing
using XSL or API and then delivered to the end user devices. For example, in the case of
graphical map representation of a specific route, an external system like Google Maps
could be utilized in the process by creating a description, for example in the Google Maps
form, from the route response.
3.4.5 Software Communications Implementation Considerations

Deliverable D2.1 Section 4.1.2 on technology and performance constraints described the software and technology used by the JPs surveyed. Such technologies are related to programming language/framework, operating system and RDBMS. Since it is unlikely that all the JPs utilised the same technologies, the various technologies engaged in some JPs are listed below.

Programming language: C++, ASP 2.0, MS IIS, Java, J2EE on JAVA SE 1.4 (JRE 1.4.2_10), Apache Tomcat 5.0.27, Apache Maven 1.0.2

Operating system: Windows server 2003, Linux

RDBMS: MS SQL-Server 2005, MySQL, Oracle, PostgreSQL 7.4.13
4. DATA FRAMEWORK

4.1 Data Architecture

This section considers the required data in order to be able to provide the functions and user services described in the previous section. Apart from WISETRIP system data, data can be provided by cooperating systems such as journey planners or booking systems incorporated within WISETRIP, as well as external information sources i.e. the user data related to selected trip, user preferences or user profile.

Information provided by the journey planners can be analysed and separated into two main sections. These are the generic data (include also JP Metadata) and the JP specific data. The generic data encapsulates the information that is required from all participating JPs in order for the WISETRIP system to function and provide the user with a uniform response. Whereas JP specific data is information provided by the different JPs that could be adapted to WISETRIP in order to enhance user experience.

Data required to establish user profile and characterise user preference is distinguished in accordance to journey planner services and limitations. The data required for a complete trip lifecycle is analysed and separated into personal trip cycle data and trip cycle model data. Personal trip cycle data being the information provided by the user that “personalizes” the WISETRIP system’s response to a user’s request. Whereas trip cycle model data consists of the information required by the WISETRIP system in order to create the personal trip cycle data and support the user’s trip (i.e. from the moment of tickets reservation until the moment of disembarkation in the destination).

WISETRIP aims to enable multi-modal trip-planning between different countries by integrating several Journey Planners. Therefore a need arises to offer multi-lingual support throughout its interface. A mechanism to perform this language adaptation within the WISETRIP core system, and the required data for its implementation, is discussed in the later sections.

Analysis of the data architecture ends up with data schema and format considerations to summarize how different travel preferences can lead to different data requirements. Within this section potential data implications provided by JPs are discussed and analysed. Data requirements and model described below, can be considered as initial rather than final one. Implementation and algorithmic decision might define changes to the proposed data, which will trigger tasks towards the refinement and detailing of the data model.

4.1.1 Data about Journey Planners – Metadata

For WISETRIP to expand across the borders of the participating journey planning system, it is necessary to acknowledge the generic information of each JP. This opens the possibility to identify differences between JPs against different countries, henceforth an appropriate approach can be made to address the potential problem.

**Technical Access Information:** For each JP, identifiers for access (i.e. URLs, IP addresses, or even authentication data) or other should be defined in order to be used practically when WISETRIP will need to access.

**International node terminal and type of terminal:** Information should contain description of geographical areas (nation names, region names, city names, map coordinates) that describe the area that the JP information supports.

**International node terminal and type of terminal:** WISETRIP focuses on international (or extra JP) journey planning, consequently the system needs distinct node terminal information. Whilst the national JPs are likely to provide the local or the national or the neighbouring countries routes network and schedules, far international routes (such as travel between UK and Greece or between Finland and China) would normally be referred
to a different JP. By identifying the type of node terminal, whether international or national/local, WIETRIP would be able to focus on its aim.

Means of transport: the traditional way of looking at different means of transport used to reach a destination is by the distance of a trip/journey. A local trip would normally be covered by walk, or cycling; a regional trip would be covered by local bus or light rail; an inter-regional (city to city) trip would be covered by regional bus/coach or train or flight; and an international trip would be covered by international bus/coach or train or flight, where overseas travel would allow only ship or long haul flight to cover the trip/journey (otherwise the trip/journey would not be economically feasible). By identifying the means of transport provided by JPs, WIETRIP would be able to identify the gap for addressing international trip/journey connections.

Type of queries: the available journey planning systems always have a use case to ask the user for starting point and destination. In the instances where WIETRIP focuses on a long international journey trip, the date and time of a planned trip/journey would be essential. Section 3.1.1 described the main components of potential user queries that can be used in WIETRIP. This includes: origin, destination, departure time, arrival time, trip duration, numbers of hops, modes of transport, preferred intermediate nodes, cost limit, number of persons, sorting preferences and special requirements. Below is the extended list of potential type of queries that can be provided by JPs (taking an example from www.transportdirect.com):

- Starting point – From (names of starting point: country, city, area, street, other),
- Destination point – To (names of destination: country, city, area, street, other),
- Date and time of planned trip/journey,
- Means of transport & preferred ones: air, rail, ferry, bus/coach, metro, ...etc, including walking mode or private car
- Find a place through address or postcode or map (identifiers of interface components such as addresses),
- Find a specific mode of transport to reach the destination (identifiers of interface components),

Additionally to those basic query data, conditions and other information can be specified within the query, such as:

- Duration of the journey (exactly, approximately, greater than, less than)
- Number of Hops (or limit of Hops)
- Preferred intermediate nodes
- Cost Limit
- Number of Persons
- Sorting Preferences (by duration, by means of transport, by departure time, by arrival time...)
- Special Requirements i.e. E&D, vegetarian, green routes

Format of response: when the query has been input by the user, the response format for JPs would be the travel itinerary. This will include information such as:

- Options for the means of travel and the itinerated time
- Walking time to a terminal node (for bus/train),
- Departure time for bus/train/flight with the service number,
- Transit time,
Approximate time to arrive at the destination,
Different level of map (i.e. street, city, intercity, international), distance and
duration and ticket/cost of travel,
Fare & Currency
Other information such as self alerts, possibilities to modify the travel

4.1.2 Journey Planner Specific data

In the previous section the generic information of each JPs was acknowledged. Each
journey planner that is incorporated in the WISETRIP system currently offers a series of
services to its users. These services may differ from JP to JP however they could be
adapted to the WISETRIP system in order to offer a broader spectrum of services to the
end user. Below is the extended list of potential information given by JPs:

- Fare & Currency Information
- Booking Capabilities
- Dynamic Travel Information
- Points of Interest
- Weather Information
- Environmental Information / CO2 Emissions

Fare Information is available for trains in the Finnish JP, journey.fi, and for multi-modal
transportation in the UK based JP www.transportdirect.info.

Booking Capabilities are offered by the Greek JP Enosis, through the ForthCRS platform
for all ferry transport within Greece. In the journey.fi JP the user can buy train tickets
through a link to VR Group website.

Dynamic Travel Information is currently not offered by all JPs. The Enosis JP provides the
user with real-time information (i.e. delays or cancellations) about departures – arrivals
from/towards the Athens International Airport from the airports of Heraklion, Santorini,
Thessaloniki, Mykonos, and Rhodes. Furthermore Enosis JP provides real-time
information about departures – arrivals of ferries at Heraklion port, as well as inhibited
ferry departures. This information is available via SMS, e-mail (according to the user’s
profile).

The list of places of interest according to the user search is another useful service offered
by some of the participating JPs (i.e. Hangzhou Bus, Journey.fi). Different point of
interest information (e.g. accommodation services, hospitals and health services,
restaurants and cafés, transportation and parking) is available free of charge by the
journey.fi JP.

Finally, weather and environmental information is offered by different JPs. For example
www.transportdirect.com offers the ability to calculate CO2 emissions for a car or public
transport for a specified journey.

www.transportdirect.com provides an automated texting service. Texting an 8-digit bus
stop reference code to the txt2traveline number 0777 608 2 608 will return the departure
times of the next few buses from your chosen stop.

An important step in defining services across the WISETRIP system is incorporating the
laws and regulations each JP is subject to. For example data protection issues arise when
considering planned trips. In the UK trip data may not be stored for more than 3 weeks.
The WISETRIP system therefore must adapt and generate appropriate notifications
regarding trips or trip segments scheduled in the UK.
4.1.3 User Profile Data

This section describes the user profile requirements which are needed to realize the personalized services provided by WISETRIP. These data requirements can be categorized into necessary, complementary and history data. The necessary requirements are the minimum user profile data required in order for WISETRIP to provide basic personalized services. The complementary requirements are additional data the user may or may not provide in order to customize the WISETRIP services to his/her needs. Finally the history requirements are data that encapsulate the user preferences for his/her travel history.

The necessary requirements: These data are required by WISETRIP for successful registration to the system. The user profile must include for each user his name, his username and password for login capability, a valid email or mobile number, and a preferred language. The email or mobile number provided will be used for notification purposes by the WISETRIP system. The user profile will provide the ability to select whether notification will be received by email or mobile or both. The preferred language option will be used throughout the WISETRIP interface i.e. query results, notifications etc.

The complementary requirements: Furthermore, the user profile must handle address settings. Home address, business address, mail address can be supplied for additional services, ticketing, booking, place of trip origin. The user could also define the method he receives notifications as being push or pull.

The history requirements: Data protection issues are important and need to be checked for each country according to the journey planners participating. Therefore, although the user profile must provide the ability to store and provide the history of completed journeys it must be in accordance to the law in each country. The user profile shall offer the ability to define the limit of stored journeys and time constraints concerning each stored journey. Finally, the user profile shall provide the option of defining periods and amount of notifications generated by the WISETRIP system concerning stored journeys.

<table>
<thead>
<tr>
<th>Profile Data Type</th>
<th>Data Required</th>
<th>Comments / Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Necessary</td>
<td>Name</td>
<td>Username, password</td>
</tr>
<tr>
<td></td>
<td>Login Information</td>
<td>For receiving SMS notification</td>
</tr>
<tr>
<td></td>
<td>Mobile Number</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Email</td>
<td>Method of notification via Email, SMS</td>
</tr>
<tr>
<td></td>
<td>Language</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Alert Notification Option</td>
<td></td>
</tr>
<tr>
<td>Complementary</td>
<td>Home Address</td>
<td>Method of obtaining information via WISETRIP services</td>
</tr>
<tr>
<td></td>
<td>Business Address</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Mail Address</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Place of Trip Origin</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Push, Pull, Both, Preference</td>
<td></td>
</tr>
</tbody>
</table>
### 4.1.4 Personal Trip Cycle Data

In this section, the data requirements which are needed to generate a personal trip cycle data will be analyzed. In order to formulate a personalized trip, certain data is required by the user as input to the WISETRIP system. These data can be used to identify the user himself or to identify his trip preferences. The first set of data were analyzed in the previous section and are defined as **user profile data**. The second set of data are drawn from user preference and can be standardized i.e. user-role, or generated specifically for a particular trip.

Considering the following scenario: the user has defined an origin and a destination point and provided both starting date and time (by default current date), and ending date and time of the trip. He can then proceed to personalize the WISETRIP system’s response by adding various parameters to his initial query.

The WISETRIP interface will allow the user to define means of transport settings. Preferred means of transport could be registered i.e. bus, train, ship, and airplane. Means can be hierarchically categorized by the user, separated to individual and public transport, or excluded from travel planning. The user will have the ability to define maximum walking distance he/she is willing to travel. Furthermore, the number of hops between origin and destination may be specified in order to formulate the personal trip. Time constraints could also be imposed on the query to WISETRIP system.

Furthermore, cost limitations could be taken into consideration whilst generating a trip cycle. Trip planning could be parameterized depending on the user preferences for ticketing and seat reservation. The user must be able to register preferred transportation companies and class when entering a query to WISETRIP. For booking purposes the user profile could include financial data. These data could include payment method, bank account, credit card details.

A time period following the disembarkation in the destination is defined as post trip phase. This period might refer to accommodation search and transfer to the selected hotel. The interface must offer the option to define the preferred hotel category, room type, or even hotel brand. Defining places of interest within WISETRIP such as restaurants, museums, could be used to enhance user experience.

The above inputs are considered by the **personal trip data generator module** while producing the notifications corresponding to a specific journey. Table 7 below summarizes the data required when considering trip cycle personalization.
### Table 7: Personal Trip Life Cycle Data

<table>
<thead>
<tr>
<th>Personal Data Type</th>
<th>Data Required</th>
<th>Comments / Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>*(Pre-)*Trip Information</td>
<td>Preferred means of transport</td>
<td>Hierarchical query based on preferred mean of transport</td>
</tr>
<tr>
<td></td>
<td>Number of hops</td>
<td>Limitation of intermediate changes</td>
</tr>
<tr>
<td></td>
<td>Time Limitation</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Price Limitation</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Maximum walking distance</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Include or exclude mean of transport</td>
<td></td>
</tr>
<tr>
<td>Booking Information</td>
<td>Transportation companies</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Transportation class</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Payment method</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Bank Account</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Credit Card Data</td>
<td></td>
</tr>
<tr>
<td>Post-Trip Information</td>
<td>Accommodation Information</td>
<td>Hotel Category, Hotel Brand, Room Type</td>
</tr>
<tr>
<td></td>
<td>Places of Interest</td>
<td>Restaurants, museums, events</td>
</tr>
<tr>
<td></td>
<td>Car Hire Data</td>
<td>Preferred Company, Car Type, Car Park</td>
</tr>
</tbody>
</table>

### 4.1.5 Trip Life Cycle Data

A trip can be considered that begins from the moment of tickets reservation until the moment of disembarkation at the destination. Up to its completion the trip passes from various distinguish and sequential phases - periods. This sequence of phases defines the Trip Life Cycle. Specifically:

- Pre-trip Phase: it concerns the time period from the moment of ticket reservation (booking) up to the moment of departure
- En-route Phase: it concerns the time period from the departure up to the moment of disembarkation in the last itinerary
- Post-trip Phase: it concerns the time period that follows the disembarkation in the destination. This period might refer to accommodation search and transfer to the selected hotel.

Each of the above phases can be divided in individual periods with short duration. In the frames of WISETRIP mainly the two first phases will be covered.

**Pre-Trip Phase**

This phase can be divided in three individual periods:

- Booking Period
Commitment Period
Waiting Period

At the first period the traveler has made the ticket reservation but has still not purchased it. During this period the traveler can make changes in the selected journey with relative easiness and without cost. For example a traveler might have made a reservation for a trip but also be on the waiting list of another more suitable route. If therefore, up to the payment deadline, the waiting list is fixed the traveler could simply change his reservation. Booking period expires with the ticket purchase.

The period from the moment of ticket purchase up to a certain time before the trip departure is the commitment period at which the traveler has committed to follow a specific route and any change is difficult to make.

The waiting period refers to the traveler waiting time in the station of first means that will be used. If the trip is constituted from more than the one segments, the traveler will also pass from other waiting periods.

En-route Phase

This phase is constituted by individual periods that interchange with the following sequence:

- Waiting period
- Boarding period
- Moving period
- Disembarkation period

The first period refers to the period of waiting in terminal stations of transport means. During this period, and accordingly with the transportation type important time points might exist that should be taken care. For example important time points for a flight are the baggage collection and the acquisition of boarding pass.

The second period concerns the time period at which the process of passengers’ embarkation to the transport means is taking place. For example in a flight it concerns the period at which the following steps are executed:

- boarding pass control
- hand luggage control
- security control
- transfer to the airplane
- finding of seat (in the airplane) and hand luggage settlement

The moving period refers to the time that the transport means is moving. Finally, the phase of disembarkation concerns the process of exit and removal from the transport means.

If a travel is consisting of more than one route then the above periods are repeated with this sequence as many times as the number of routes.

For each phase of the Trip Life Cycle Model the data that will be distributed to the user (taking also in mind the user’s personal preferences) should be defined. Moreover for each period of the Trip Life Cycle potential extraordinary information that might concern the users should be identified as well as the source from which this information will be available.

The following table presents all scheduled but also potential extraordinary information that is available for travelers at each phase of the Trip Life Cycle Model described earlier. The trip life cycle model is actually defining the rules, on which the Personal Trip Data
Generator and Personal Trip Scheduling will be based in order to generate and define Personal Trip data.

Table 8: Trip Life Cycle Information

<table>
<thead>
<tr>
<th>Period</th>
<th>Scheduled information</th>
<th>Extraordinary information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Commitment</td>
<td>• Reminder of trip details</td>
<td>• Notification for availability in alternative route (waiting list)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Notification for import of unscheduled route</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Notification for a route change</td>
</tr>
<tr>
<td>Waiting</td>
<td>• Reminder of details for the ticket control</td>
<td>• Notification for delay/cancellation of a route</td>
</tr>
<tr>
<td></td>
<td>• Reminder of boarding details</td>
<td>• Notification for change in the boarding process (time, exit)</td>
</tr>
<tr>
<td></td>
<td>• Information related to activities in the waiting area</td>
<td>• Notification for changes in next routes</td>
</tr>
<tr>
<td>Boarding</td>
<td>• Reminder of boarding rules</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Reminder of seat number</td>
<td></td>
</tr>
<tr>
<td>Moving</td>
<td>• Information relative with the trip</td>
<td>• Notification for events that affects the smooth progress of the trip</td>
</tr>
<tr>
<td></td>
<td>• Reminder of expected arrival time</td>
<td>• Notice for changes in next routes</td>
</tr>
<tr>
<td></td>
<td>• Reminder of details for the next route</td>
<td></td>
</tr>
<tr>
<td>Disembarkation</td>
<td>• Information for the next route</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Information for departure from the transport station</td>
<td></td>
</tr>
</tbody>
</table>

4.1.6 Linguistic Data

WISETRIP intends to be an integrated interface for several Journey Planners, which will enable multi-modal trip-planning between different countries. Therefore it is trying to be an international tool and, as such, multi-lingual support should be available.
The WISETRIP interface will allow the user to select his/her preferred language, which will be the language used for any communication to and from the user during the whole process (alerts, indications, warnings, etc). The user will be able to make this selection either at the query phase or whenever he/she creates or modifies his/her profile.

Whenever a query is made, in a previously specified language, the system will forward that query to the corresponding JP, together with information about the language in which the query is being submitted and the language in which the response is preferred. Once the query has been made in the appropriate JP:

- a) If the JP can answer in the selected language the response is given in the correct form and no translations will be needed.
- b) If the JP can not provide the response in the selected language, a translation will have to be done through the language adaptation function.

To perform language adaptation the following data will be necessary:

- User’s preferred language (information of the user profile)
- Response from the JP (to be translated)
- Response language
- Dictionary Database (to perform the translation)
- Icons Data

As far as concerns the dictionary database, it will just be limited to terms, field names and message templates of WISETRIP system. Instead of building all combinations of pairs for dictionaries, for all languages, it might be wiser to build only dictionaries based on a reference language, that in our case it should be English. For example, instead of building a Swedish-Greek dictionary, it will be easier to have a Swedish-English-Greek dictionary.

4.1.7 Other Data – User Context – Location based

The WISETRIP system is going to offer to the end users user context and location-based data. The context can be defined as the surroundings, circumstances, environment, background or settings which determine, specify, or clarify the meaning of an event (source: ISO Workshop on address standards: Considering the issues related to an international address standard 25 May 2008, Copenhagen, Denmark, ISBN 978-1-86854-689-3 “Ubiquitous Geographic Information (UBGI) and address standards” by Sang-Ki Hong, Anyang University, Korea.). So in the WISETRIP system the user context is based on and derives from the user’s personal profile & preferences and the present location of the user. Naturally the end user device will also affect to the data, at least to the way it will be presented to the user. Thus the provision of user context data includes from the user an indication of an interest in receiving data based on his profile and current location.

Storing and retrieving user context data includes a system and method for storing data in a network device (here the WISETRIP server) that is connected to the end-user device and allows for the secure and on-going storage of the needed user context data without adding unnecessary complexity to existing web server methods, while at the same time supporting redundancy and facilitating effective load-balancing. By storing user context data in the WISETRIP server, the invention reduces the complexity of the system and increases the overall scalability of the system. This approach makes it also possible to provide the SMS-based service users with location-based user context data.

The user context data provision means that WISETRIP needs to be able to reliably identify the user’s context automatically e.g. in order to offer (location-based) information relevant to the present trip-cycle and based on the user’s profile. To make this happen relevant user context sensors (position and time detection for instance by
GPS) need to be identified, user context information needs to be collected (from user profile and all possible external data sources), represented, analyzed in many ways (by different devices and user interfaces), etc.

There are several issues related to user context data such as:

- Representation of context and observation data, mining data from user context representations
- Visualization of observations context of the user
- Management of repositories of context and observation data
- Personal information management and context data
- Extending user profile modeling
- Privacy and security issues.

There are some challenges that need to be taken into account and resolved such as:

- How to observe the user to detect and capture context? What sources and sensors are needed and feasible? What types of (interest) indicators can be inferred from that? How can one represent the observations initially?
- How can user observations and user context models be represented? What is an appropriate format to exchange observed user-related data? How can data be represented? Do certain types of data reveal any special qualities?
- How can – through analysis of observation data – further valuable information be inferred? Is there a way to detect processes or task patterns from observations and context? Can observed data be used for detecting similarities between tasks? Is there a way of inferring user-related information from observed contextual data? How can diverse observations be merged such that expressive structural information is revealed?

### 4.1.8 Data Schema and Format Considerations

Section 3.4.3 described how different travel preferences can lead to different data requirements. At the same time, for local travel within short distance, travellers would not normally plan their journey using JP services; instead they often make a spontaneous decision to walk, to cycle or board local public transport. For a medium distance travel, travellers may use JP services for choosing available public transport services supported with time schedule and fares information. Otherwise, travellers may also use JP services to find a route for private car travel. For this medium distance journey, people may make a decision about their journey depending on time and cost of travel. For a long distance journey travellers would certainly make a considered decision to select their preferred journey. This is especially true because of the time and cost risk encountered for such a journey. For this reason WISETRIP has to provide a robust journey planning system to support all the necessary data possible to be delivered for users to make a considered decision on their preferred journey.

Two distinctive journeys can be introduced here, one is by private car and the other one is by public transport. Whilst it is easier to measure the time and cost for a private car journey, for a public transport journey the time and cost of travel incorporate more complex data integration. For this reason, the data scheme for a journey has to be divided into two distinct groups: private car (Figure 8) and public transport (Figure 9).
Figure 8 and Figure 9 shows comparison of potential data implications that can be provided for different type of trip or journey, by private or public transport. The comparison shows that a private car trip/journey has far less data that needs to be provided by JPs whilst a public transport trip/journey implicates more data. The circle as shown in Figure 8 and Figure 9 shows the potential degree of importance for provided information as perceived by user. Data such as points of interest, CO₂ emission, trip/journey distance and so forth can be considered as secondary, than the data within the inner circles. WISETRIP focuses on the international journey planning system, so it is likely that the user would choose public transport scheme to plan their journey.
5. PHYSICAL ARCHITECTURE

5.1 Servers
The hardware for the realisation of the WISETRIP system would mainly be necessary to host the following server functionality:

- Web server
- Database Management Software server
- Application Services Server (to host main core server functionality of WISETRIP)

It is most likely that a single machine or two server machines will be enough for the demonstration phase of the project. However, to achieve scalability, and depending on the computational needs of the engaged algorithms, more servers might be added. Replication and server redundancy will be also envisaged, which will impose adjustments to the physical architecture of WISETRIP.

It will not be necessary to change the existing physical architecture at the participating JP level.

To achieve better clarity and scalability it might be useful to consider setting up a dedicated server to host the Web-services platform. It is likely that the server hosting the Web-services platform will be under the biggest load in the process and so, if needed, it would be simplest to duplicate or even multiply the capacity of only this particular server. In that case it probably would not be necessary to increase the capacity of other servers.

Potential server environment will be as follows:

MICROSOFT ENVIRONMENT
- Development Platform: Personal Computer (PC)
- Operating System: Microsoft Windows (Server 2003 or XP)
- Web Server: Internet Information Server (IIS)
- Database Management System: SQL Server
- Access to Database: through TCP/IP, using ADO (ADO.NET) or ODBC drivers
- Communication among WISETRIP modules: TCP/IP, sockets, web services or linked DLLs
- Programming Language: C++ and C#
- Programming Tool: Microsoft Visual Studio

LINUX ENVIRONMENT
- Development Platform: Personal Computer (PC)
- Operating system: Linux
- Web Server: Apache (http://httpd.apache.org)
- Relational Database Manager: MySQL or PostgreSQL
- Application Server: JBoss AS 4.2
- Programming language: Java, J2SE 5.0,
- Programming Tool: Eclipse Europa

---

3 JBoss AS 4.2 is a Java EE 1.4 application server, but Enterprise JavaBeans 3.0 is deployed by default. It requires the Java Development Kit version 5. Tomcat 6 is the Web Container bundled with it.
Both environments could be used if Core System and Personalization System are separated in two distinct server platforms.

5.2 Databases

Generally there are five main databases in the business market that may be considered: Oracle, SQL Server, Sybase, PostgreSQL and MySQL. A physical data model on each implementation would be significantly different. The underlying OS requirements will vary according to the RDBMS. For example SQL Server only runs on Microsoft Windows operating systems, while Oracle and MySQL can run on Solaris, Linux and other UNIX-based operating systems. Final RDBMS selection will have to be made in accordance to the requirements the proposed database must fulfil.

5.3 Network

All databases, application servers and web servers of WISETRIP system will by hosted at Forthnet Data Center in Athens. Conditions should secure high availability and performance, based on powerful server engines, redundant data server infrastructure and high capacity networking. WISERIP role, as a reference point for Journey Planners who can link together and co-operate within the framework of WISETRIP, requires the best possible quality of service in terms of availability, reliability, security and performance. Henceforth, the ICT infrastructure of a leading network and e-services provider will be utilized.

5.4 Devices

Equipment for interacting with the service centre such as PC (Personal Computer), in-vehicle terminal, smart phone, PDA (Personal Digital Assistant), low-technology mobile phones etc. will be necessary. These devices, at the user sites (either fixed or mobile) are also part of the physical architecture. They are connected either over the Internet (mobile and fixed IP technology) or via GSM network (messaging functionality for SMS alerting or even call back functionality). As technology evolves rapidly in the mobile devices domain, it is wise not to predetermine the type of devices to experiment the WISETRIP service. However, in principle, both low-technology devices and advanced state-of-the-art devices should be used within the validation and demonstration phases of the project.

Without these devices WISETRIP cannot fully provide the services. These devices, at the user sites are also part of the physical architecture. They are connected either over the Internet or via GSM network. The use of different devices (even by the same user, dependant on the status of the trip cycle) makes it possible to offer seamless services to the end users. As technology evolves rapidly in the mobile devices domain, it is wise not to predetermine strictly the type of devices to experiment the WISETRIP service. Both low-technology devices and advanced state-of-the-art devices are needed and will be used.

The information and services will be delivered to the end user devices either in HTML format (in the case of a www browser) or in WML format (in the case of a wap-enabled phone). The case of delivering the requested data to the user on a single format without having to transform it in HTML of WML may be considered. This can be done using, for example, XML plus XSLT style sheet for different HTML output devices (the use WML need to/will be analysed, whether it is possible to use one format). The client hardware will define the software to run on it. SMS will also be used (when the user has a low-technology mobile phone).
5.5 Implementation Considerations
Possibly, implementation of Core System might rely on different technological environments than the Personalization System. There will not be any problem implementing these two systems in two different technologies as long as communication between the two systems is implemented using Web Services. However there might be some bottlenecks that need to be considered:

- A common database is required that will store all data required by the WISETRIP system to function. Two separate databases may require synchronization between them which will produce unnecessary complications to the system design.
- Therefore a decision will be required, as to which server will host the database
- Access lists have to be made in order for the systems to communicate.
6. FINDINGS ON RISKS AND CONSTRAINTS

The proposed methodological framework for WISETRIP development is the design of the system which will be based on identifying and integrating the passenger information and journey planning functionality that best fit to the user needs. A detailed analysis on risks and constraints of WISETRIP is described in D.2.1 Section 4.

The biggest risk is that existing Journey Planners do not ‘connect’ to the WISETRIP portal. As the responsibility for developing the server-side code (which enables interfacing with the WISETRIP client-side web services) lies with the existing JP developers, this presents the considerable risk that there will be only limited geographical coverage. This in turn weakens the offer of an international JP service. To avoid this, the WISETRIP service needs to provide sufficient added value to the existing JPs to encourage their participation.

As described in D.2.1 Section 4.3.4 on the consideration of opportunities and risks, the most important opportunity for WISETRIP arises from the evolution of mobile Internet services either over 3G or over wireless broadband technology (WiFi, WiMax, etc.). It is also described that the most important risks born from this project would be:

- The non-access by the most important transportation means or countries will constitute a factor of non-completion of content and it will have a result of a system lacking the characteristics of information completeness and geographical coverage.
- The expected existence of competitive efforts (either from private initiative or from the public sector) that are based in mechanisms of public information (web sites), in-vehicle systems, voice systems of transport companies, ports etc.

As regards the constraints, again extracting from the report of D.2.1 Section 4 about system constraints and limitations, it was described that there are three categories: technological, institutional and business constraints. The technological constraints include the difficulties in integrating functionalities and technologies offered by different participating journey planners. The institutional constraint is concern with national, European and international issues on personal data protection. The third constraint is about the business oriented development.

Additionally, since WISETRIP focuses on international journey planning systems, some of the further challenges to be addressed are:

- Not equally supported data by the participating JPs.
- Incapability to incorporate local JPs with international travel agent for train/flight tickets/fares.
- Language translation barrier for exchanging JPs data across national border.
- Inconsistency of the type of different available data from JPs. For instance: some JPs provide fares, some not, therefore the cost for the complete journey won’t be addressed.
- Level of services for different JPs.
# 7. WISETRIP GLOSSARY

<table>
<thead>
<tr>
<th>Active Services</th>
<th>Information Push Services</th>
</tr>
</thead>
<tbody>
<tr>
<td>Airplane</td>
<td>Means of transporting freight or passengers by air</td>
</tr>
<tr>
<td>Alerts</td>
<td>Is a service offered to individuals by search engine companies which notifies its users by email, SMS, about the latest information regarding their requested journey(s)</td>
</tr>
<tr>
<td>AP</td>
<td>Access Point</td>
</tr>
<tr>
<td>API</td>
<td>Application Programming Interface</td>
</tr>
<tr>
<td>B2B</td>
<td>Business to Business</td>
</tr>
<tr>
<td>Booking</td>
<td>A reservation in the travel and tourism industries</td>
</tr>
<tr>
<td>Bus</td>
<td>A large road vehicle designed to carry passengers in addition to the driver.</td>
</tr>
<tr>
<td>Cancellation</td>
<td>Notification that requested trip or journey will not take place.</td>
</tr>
<tr>
<td>Car Hire</td>
<td>An agency or company that rents automobiles for periods of time. Can be arranged through various websites</td>
</tr>
<tr>
<td>Coach</td>
<td>A Large Road vehicle designed to carry passengers and goods/luggage in addition to driver. Usually more luxurious than a bus, no standing allowed everyone must have a seat.</td>
</tr>
<tr>
<td>Credit Card</td>
<td>A card that allows people to buy items without money. It charges the money needed to their bank account, so the person will pay later.</td>
</tr>
<tr>
<td>Cycling</td>
<td>Use of bicycles (human powered vehicles)</td>
</tr>
<tr>
<td>Delay</td>
<td>Information advising passenger that they will have to be aware that there is a problem that will cause potential of missing connections or arriving late.</td>
</tr>
<tr>
<td>Disabled Access</td>
<td>Means of highlighting suitability of services for people with disabilities</td>
</tr>
<tr>
<td>Diverted</td>
<td>Information sent to passenger advising that due to various circumstances (weather, technical problems etc) the journey will not be arriving at the stated destination or arrival time and that alternative route/destination is required</td>
</tr>
<tr>
<td>DRT</td>
<td>Demand Responsive Transit</td>
</tr>
<tr>
<td>Term</td>
<td>Definition</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Exact Fare</td>
<td>No change will be given by the driver of the vehicle</td>
</tr>
<tr>
<td>Express Service</td>
<td>Limited Stop Service</td>
</tr>
<tr>
<td>Extra Journey Planner Segment</td>
<td>The segment, linking two intermediate locations of the journey, which has no common hops with the Origin Journey Planner Domain and the Destination Journey Planner Domain.</td>
</tr>
<tr>
<td>Fare(s)</td>
<td>Cost of travel</td>
</tr>
<tr>
<td>Ferry</td>
<td>Means of water-based transportation for freight and passengers</td>
</tr>
<tr>
<td>Frequency</td>
<td>Definition of how often services operate every 30 minutes</td>
</tr>
<tr>
<td>Galileo</td>
<td>Global navigation system built by the European Union (EU) and the European Space Agency (ESA)</td>
</tr>
<tr>
<td>GIS</td>
<td>Geographical Information System</td>
</tr>
<tr>
<td>GSM</td>
<td>Global System for Mobile</td>
</tr>
<tr>
<td>GPS</td>
<td>Global Positioning System</td>
</tr>
<tr>
<td>GPRS</td>
<td>General Packet Radio Service</td>
</tr>
<tr>
<td>GUI</td>
<td>Graphical User Interface</td>
</tr>
<tr>
<td>HST</td>
<td>High Speed Train</td>
</tr>
<tr>
<td>HTTP</td>
<td>Hypertext Transfer Protocol – World Wide Web protocol</td>
</tr>
<tr>
<td>HTTPS</td>
<td>Hypertext Transfer Protocol Secure</td>
</tr>
<tr>
<td>Icon</td>
<td>Means of depicting transport modes and information</td>
</tr>
<tr>
<td>IEEE</td>
<td>Firewire interface serial bus interface standard</td>
</tr>
<tr>
<td>Interchange point</td>
<td>An intermediate location, where the user should disembark from a specific means of transport and get on board other means of transport, either of the same or different mode</td>
</tr>
<tr>
<td>International Segment</td>
<td>An international journey that is necessary within the total journey solutions that are available for the user</td>
</tr>
<tr>
<td>Internet</td>
<td>Worldwide, publicly accessible series of interconnected computer networks</td>
</tr>
<tr>
<td>IP</td>
<td>Internet Protocol</td>
</tr>
<tr>
<td>ISO</td>
<td>ISO image is an archive file</td>
</tr>
<tr>
<td>ITS</td>
<td>Intelligent Transport Systems</td>
</tr>
<tr>
<td>Journey (or JOURNEY SOLUTION)</td>
<td>Description of the total travel requirements from start to finish. May comprise of one or more trips using one or more modes</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>JP</td>
<td>Journey Planner</td>
</tr>
<tr>
<td>LAN</td>
<td>Local Area Network</td>
</tr>
<tr>
<td>Local Service</td>
<td>Service provided within local area linking towns/cities and areas within towns/cities and villages.</td>
</tr>
<tr>
<td>Local Train</td>
<td>Usually stops at more frequent periods resulting in longer journey times for passengers travelling longer distances</td>
</tr>
<tr>
<td>Messages</td>
<td>Means of communication SMS, email, GPRS, between passengers informing on updated situation regarding status of travel</td>
</tr>
<tr>
<td>Metro</td>
<td>A railborne vehicle lighter than a train, designed to carry passengers within, close to, or between villages, towns/cities, primarily on streets or underground</td>
</tr>
<tr>
<td>Mode</td>
<td>Type of Transport Available, Train, Bus, Coach, Taxis, Plane, Ferry etc.</td>
</tr>
<tr>
<td>MRT</td>
<td>Mass Rapid Transit</td>
</tr>
<tr>
<td>Operator</td>
<td>Name of organisation providing the service</td>
</tr>
<tr>
<td>Origin (Destination) Journey Planner Domain</td>
<td>The set of segments resulting from a single Journey Planner and corresponding to partial journeys starting from the Origin (Destination) location, reaching upto an intermediate location</td>
</tr>
<tr>
<td>Passive Services</td>
<td>On Demand – Information Pull Services</td>
</tr>
<tr>
<td>Payment</td>
<td>Method required to purchase ticket for travel</td>
</tr>
<tr>
<td>PC</td>
<td>Personal Computer</td>
</tr>
<tr>
<td>PDA</td>
<td>Personal digital assistant</td>
</tr>
<tr>
<td>POI</td>
<td>Points of Interest</td>
</tr>
<tr>
<td>Pull</td>
<td>Means for users to extract travel information from WISETRIP</td>
</tr>
<tr>
<td>Push</td>
<td>Means of WISETRIP system to automatically send updates to user</td>
</tr>
<tr>
<td>PT</td>
<td>Public Transport</td>
</tr>
<tr>
<td>Query Response</td>
<td>It is the list of journey solutions that satisfy the user requirements</td>
</tr>
<tr>
<td>Term</td>
<td>Definition</td>
</tr>
<tr>
<td>-------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Real Time Information</td>
<td>Means of reporting on actual performance of multi-modal transport systems</td>
</tr>
<tr>
<td>Route</td>
<td>Defined set of roads, rails, directions to be used to travel between points</td>
</tr>
<tr>
<td>Re-Route</td>
<td>Deviation of intended direction due to various circumstances such as road works, engineering works, weather, technical problems etc</td>
</tr>
<tr>
<td>Request</td>
<td>Method of establishing information regarding your personal requirements</td>
</tr>
<tr>
<td>Service</td>
<td>The name or number of the mode chosen to travel</td>
</tr>
<tr>
<td>Season Ticket</td>
<td>Method of payment covering more than 1 days travel</td>
</tr>
<tr>
<td>Segment (or Partial Journey)</td>
<td>A part of a journey that corresponds to a specific move from an intermediate location (or even the Origin) to another intermediate location (it could be the destination)</td>
</tr>
<tr>
<td>Smart Card</td>
<td>Any pocket sized card with enabled integrated circuits which can process information.</td>
</tr>
<tr>
<td>SMS</td>
<td>Short Message Service</td>
</tr>
<tr>
<td>Stage</td>
<td>Public Transport industry term used to describe point where fare scales may adjust based on distance or zones.</td>
</tr>
<tr>
<td>Taxis</td>
<td>Is a type of public transport for single user, or small group of passengers, typically for a non-shared ride.</td>
</tr>
<tr>
<td>Taxi Rank</td>
<td>Location where Taxis are available</td>
</tr>
<tr>
<td>TDC</td>
<td>Travel Dispatch Centre</td>
</tr>
<tr>
<td>Terminal</td>
<td>Point where transport can be accessed usually a building such as a Railway//Bus/Coach Station, Airport or Ferry Terminal</td>
</tr>
<tr>
<td>Ticket</td>
<td>Proof of purchase for individual to travel</td>
</tr>
<tr>
<td>Train</td>
<td>A connected series of vehicles that move along a track to transport freight or passengers from one place to another</td>
</tr>
<tr>
<td>Tram</td>
<td>Tram, Tramcar, trolley, trolley car or streetcar is a railborne vehicle lighter than a train, designed to carry passengers within, close to, or between villages, towns/cities, primarily on streets</td>
</tr>
<tr>
<td>Trip (or Hop)</td>
<td>Individual movement between 2 points</td>
</tr>
<tr>
<td>UI</td>
<td>User Interface</td>
</tr>
<tr>
<td>Term</td>
<td>Definition</td>
</tr>
<tr>
<td>--------</td>
<td>------------</td>
</tr>
<tr>
<td>UML</td>
<td>Unified Modelling Language</td>
</tr>
<tr>
<td>UMTS</td>
<td>Universal Mobile Telecommunications System</td>
</tr>
<tr>
<td>URL</td>
<td>Uniform Resource Locator</td>
</tr>
<tr>
<td>User Query</td>
<td>It is the initial request of the user for trip information for a specific journey</td>
</tr>
<tr>
<td>Walking</td>
<td>Time taken for the average person to walk between modes or from modes to end destination</td>
</tr>
<tr>
<td>WAP</td>
<td>Wireless Application Protocol</td>
</tr>
<tr>
<td>WEB</td>
<td>World Wide Web is a system of interlinked hypertext documents accessed via the internet</td>
</tr>
<tr>
<td>WEB Browser</td>
<td>Software application which enables a user to display and interact with text, images, videos, music and other information typically located on a Web page</td>
</tr>
<tr>
<td>XML</td>
<td>Extensible Markup Language</td>
</tr>
</tbody>
</table>