Theoretical and Empirical Researches in Urban Management
Volume 8 Issue 4 / November 2013

APPLYING THE CoReS REQUIREMENTS DEVELOPMENT METHOD FOR BUILDING IT TOOLS FOR URBAN MANAGEMENT SYSTEMS: THE URBANAPI PROJECT

Zaheer KHAN
University of the West of England, Coldharbour Lane, BS16 1QY, Bristol, United Kingdom
Zaheer2.Khan@uwe.ac.uk

David LUDLOW
University of the West of England, BS16 1QY, Bristol, United Kingdom
David.Ludlow@uwe.ac.uk

Wolfgang LOIBL
Austrian Institute of Technology, Giefinggasse 6, 1210, Vienna, Austria
Wolfgang.Loibl@ait.ac.at

Abstract

Gathering and managing requirements from stakeholders is an essential task for the successful development of IT tools for any application domain. However, specifying stakeholders' requirements for a collaborative research and development project in the urban management domain is especially challenging. The reason for this is that stakeholders are drawn from different professional, scientific and national backgrounds and contexts, and therefore have various and differing research, technological and application domain-specific objectives. These challenges make it difficult to identify commonalities in requirements that can result in the development of generic IT tools which can be applied in other cities. These various challenges make it highly desirable therefore to identify and apply a coherent methodological approach to the management of the requirements gathering process and stakeholder engagement more generally.

In this paper, we introduce a coherent methodology and a generic requirements engineering process, namely – 'CoReS – Collaborative Requirements Engineering and Stakeholder engagement'. This process is applied to the UrbanAPI-project - a collaborative research project, in which eleven partners from six European countries are collaborating to develop IT tools to support policy making, urban planning and participatory governance at different urban scales. The CoReS method results in the identification of commonalities in stakeholder requirements from four major cities located in different EU member states with the objective to develop generic IT tools and applications.

More specifically, this paper reports on the urban planning issues and needs for generic IT tools. In addition, it presents strengths and weaknesses of the CoReS method applied in the UrbanAPI project. Furthermore, it is argued that this experience can effectively support the specification of a roadmap in defining the requirements for the development of IT tools for decision support in the urban planning domain, that can be applied to a wide range of cities throughout Europe.

Keywords: Requirements Engineering, Urban Management, IT Tools Development, Stakeholder Engagement

1. INTRODUCTION

1.1. Application domain – urban management

Democracies have been challenged in recent times by increasing demands from citizens to take part in decision-making, for example infrastructure decisions and planning, especially at local and regional levels (Dahl 1998; Held 2006). As a result, policy making now aims to become a process integrating top-down activities carried out by urban planning agencies, with bottom-up elements driven by local citizens seeking to be involved in the decision making processes. The development of IT tools supporting urban management focuses on information and intelligence needs within a well-defined policy making process (Khan et al 2012), consisting of different procedural stages including: issue identification, agenda setting with consideration of alternative development options, analysis, negotiation and decision making, implementation and finally evaluation as depicted in Figure 1 (Kraemar et al 2013; Khan et al, 2012).

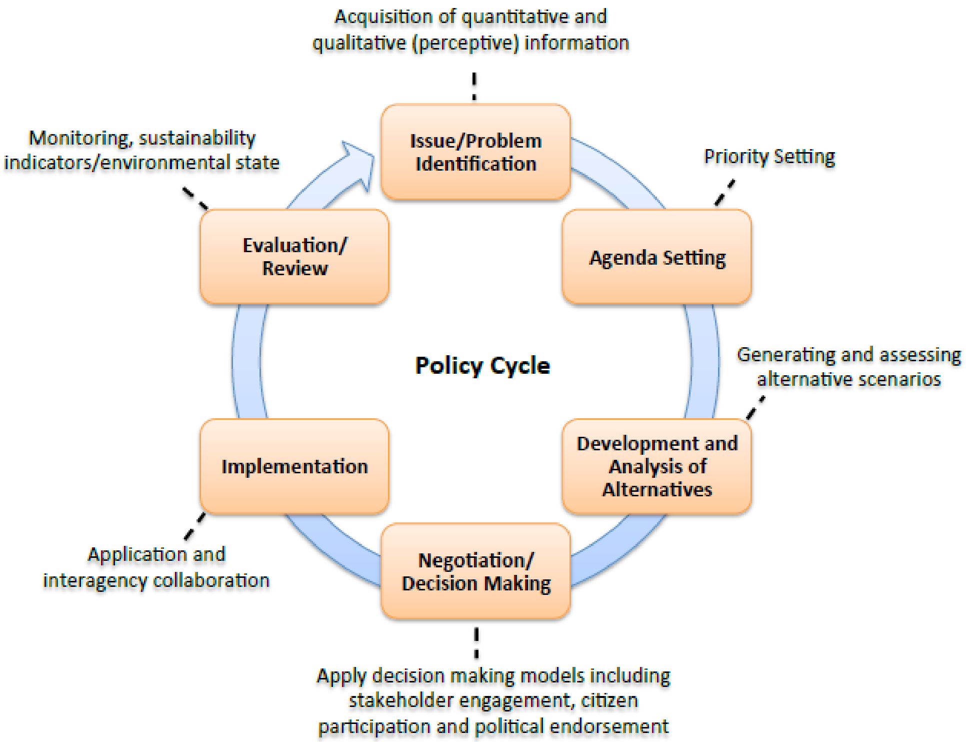


FIGURE 1 - A GENERIC POLICY MAKING MODEL (ADOPTED FROM: KRAEMER ET AL 2013; KHAN ET AL. 2012)

In this framework, IT-enabled policy development and decision-making for urban management and governance requires the involvement of various participants including citizens, stakeholders, practitioners and policy makers at different stages of the policy cycle, issues central to the objectives of

Theoretical and Empirical Researches in Urban Management
Volume 8 Issue 4 / November 2013

Theoretical and Empirical Researches in Urban Management
Volume 8 Issue 4 / November 2013

the FP7 UrbanAPI project – "Interactive Analysis, Simulation and Visualisation Tools for Urban Agile Policy Implementation" (UrbanAPI project, 2011-2014) [1].

1.2. Problem Context: Why is it challenging?

One of the prime challenges in the development of IT tools for urban management is the collection and analysis of different stakeholders' requirements. However, this requirements gathering and analysis process is not straightforward due to a number of factors including:

- defining the requirements of a range of stakeholders with different professional interests in the solutions and often from different cities and countries (i.e. collaborative R&D projects);
- identifying multiple and sometimes conflicting application specific, policy related and user-defined requirements;
- detecting multiple and often conflicting technological, research and development objectives, and
- recognising commonalities among various stakeholders' requirements and needs in order to develop generic solutions.

Urban management and governance systems and their operational needs, derived from the stakeholders, form the essential platform to support the system development life cycle (SDLC) including the specification of requirements, analysis and design, implementation, testing, deployment and validation of IT tools. Here Figure 2 depicts a typical system development life cycle in which user needs are identified by various governmental and non-governmental stakeholders in a particular city 'X'. However, when the system requirement specifications for the same application are developed in another city "Y", perhaps located in a different country, inevitably some different requirements will be identified due to different local political priorities and agendas. In our case the challenge is to perform rigorous analysis to identify commonalities between these requirements in order to design generic features in the IT solutions which can be applied in other cities.

Failure to recognise and respond to these challenges at the requirements gathering and analysis stage of the SDLC, results in the failure of the IT development team to incorporate the required application features. Poorly specified functional and non-functional attributes of the tools and services developed, significantly undermine the suitability of the solutions presented, which are, as a consequence, not

[1] The UrbanAPI project is funded under the European Commission's Framework Programme 7 for the duration of three years (September 2011 – August 2014).

widely adopted by cities. In addition, this often results in increasing costs of IT tool development and maintenance, for example as a result of reworking error corrections, etc (IBM Requirements Management, 2009; Cysneiros et al 2001; Henderson 2006).

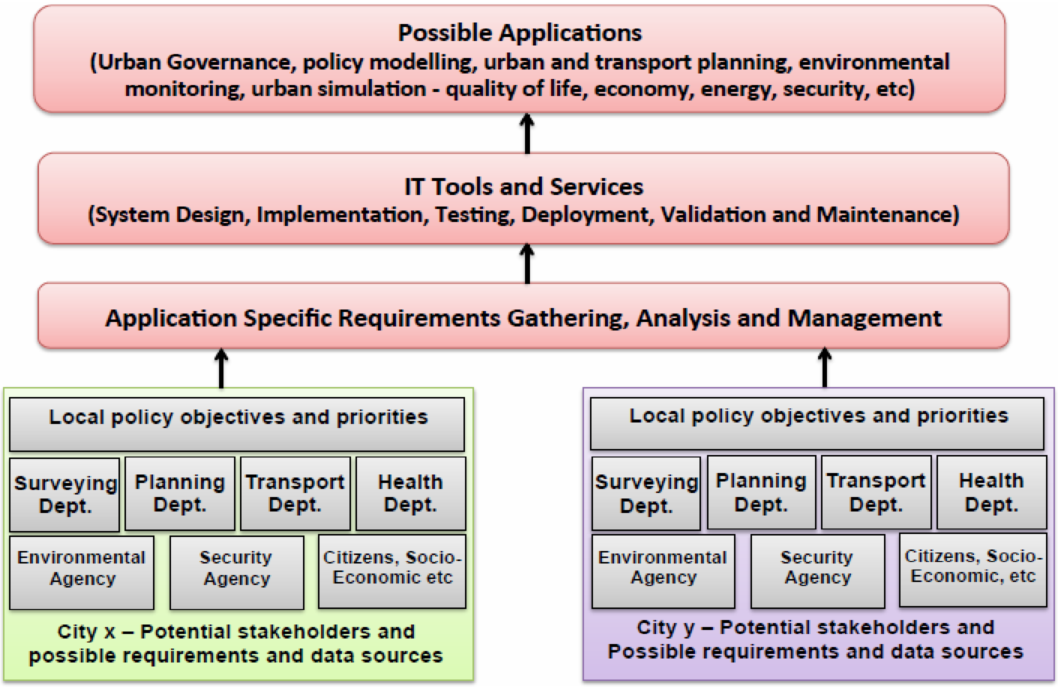


FIGURE 2 - SYSTEM DEVELOPMENT LIFE CYCLE

1.3. Research Question and Methodology

In the context of IT enabled urban governance, this paper attempts to address and answer the research question "to what extent does a generic requirements engineering process usefully support the identification of common user defined requirements for the development of generic IT-enabled urban policy-making, planning and collaborative decision-making solutions applicable to many cities?". In order to respond to this question, a user deriven requirements gathering process - the CoReS method - is introduced in this paper and applied to a real case study - the EU funded FP7 UrbanAPI research project. UrbanAPI aims to develop three generic IT applications to support urban management by collaborative decision-making and public engagement in the planning process. The development objective of the UrbanAPI project is to go beyond the existing state-of-the-art technological solutions in order to deliver attributes essential for urban planning decision-making tools in compliance with local city needs, particularly in relation to ensuring effective engagement with stakeholders including citizens (Davies et al 2012; Poplin 2012; Hanzl 2007). As a consequence, the application of CoReS method is an attempt to derive common application specific requirements from different public administrations in four European cities (Vienna, Bologna, Vitoria-Gasteiz and Ruse) and to develop generic IT solutions.

Theoretical and Empirical Researches in Urban Management
Volume 8 Issue 4 / November 2013

1.4. Paper Organisation

The structure of remaining part of this paper is as follows: in section 2, urban management requirements and stakeholder perspectives are presented, followed by introduction of the UrbanAPI project in section 3. Section 4 specifies the CoReS methodology and is explained through the UrbanAPI case study by identifying the major stakeholders in the requirements gathering process. In section 5 we discuss the outcomes of the CoReS process for UrbanAPI, and reflect on its usefulness in the urban management domain. Finally, section 6 provides some conclusions and offers an outlook on future research needs.

2. URBAN GOVERNANCE: REQUIREMENTS, STAKEHOLDER ENGAGEMENT AND IT TOOLS

Using a suitable requirements engineering method is fundamental to software development, applied in this case to the manner in which the IT community seeks to find a secure basis for the development of IT tools in the urban management domain. Since the goals of urban management are by their nature broad, varied and complex, its delivery is heavily reliant upon the action of a plurality of actors across different operationally inter-dependent policy sectors (Wong 2011). The context in which requirements are specified for urban management is therefore highly diverse and dynamic. Typically, a requirements development process consists of requirements elicitation, gathering, analysis, validation and management activities. However, in the absence of a well-defined requirements development process, IT tools for urban management may fail because they cannot fulfill the necessary stakeholder requirements and needs. In this respect the necessary dialogue with the stakeholder communities, and the subsequent process of software development raises some fundamental questions regarding user requirements in the urban management domain. In this regard, both user and technological perspectives are presented below:

2.1. User Perspectives on Urban Management

Urban management requirements, mostly defined by urban planners, seek to harness the full benefits of IT enabled urban governance (Montgomery et al, 2003; Ruble, et al. 2001; Eckhardt and Elander 2011; Kearns and Paddison 2000). These benefits are sought in order to manage the complexity of urban systems in responding to political objectives supporting sustainable urban development, ensuring an appropriate balance of socio-economic and environmental objectives defined in respect of land use management, and to fully engage with urban stakeholder communities in this process (Davies et al 2012; Poplin 2012; Hanzl 2007; Misuraca et al 2011).

In this regard a notable shift has been observed from "government" to "governance" effected by the reforms of administrations, of public management and improved regulations. As a consequence the dominant conception of the relationship between the state and the citizen have been transformed. For instance, forms of engagement with stakeholders by the state have evolved from an emphasis on top-down hierarchical models towards networked models, from steering and directing society to contextual steering and incentive provision (Rhodes 1997; Pierre & Peters 2000; Kingston R 2007). An important driving force in this transformation concerns the information overload arising from the complexity of urban systems management, experienced in both political and technical management. This has reinforced the understanding that traditional planning methodologies are outdated, and has highlighted an increasing need for tools to support the involvement of the public in decision-making, and to assist in citizen assessment of the impacts of policy-making, creating enhanced intelligence, and applied in both the management of urban complexity as well as enhancement of democracy (Felt & Wynne 2007).

Urban governance IT tools and applications have developed rapidly and in a variety of different ways ranging from direct democratic experiments such as planning cells, public assessment exercises and moderated discussions of various kinds, to experiments with the science/policy interface and impact assessment (Liberatore & Funtowicz 2003; Boyd & Chan 2002). Citizens have incorporated a number of roles in these experiments including active and participative, critical in the oversight of politics and administration, productive in generating data useful for assessing public services, and efficient in co-producing public services (Rowe and Fewer 2005). Whilst this is clearly a demanding set of roles, there is substantial evidence that participatory decision-making in the framework of urban governance initiatives has the potential to widen and deepen democratic decision-making and public service delivery (Wampler and McNulty 2011).

In this challenging context of dynamic, diverse and complex urban environments, the central question concerns the means by which user requirements can be most effectively specified in supporting IT enabled urban governance. The answers to this question almost certainly emphasise the degree of difficulty, and the great variety of solutions required. Nonetheless, there are many reasons, given the evident potential of IT innovation, to expect that IT enabled urban governance can secure effective solutions that fully meet user requirements. Furthermore, major opportunities exist for the development of common interoperable systems given the commonalities of the urban planning systems that require communication between agencies at different levels of governance from local to EU, and at the local level in a horizontal inter-sectoral perspective.

Theoretical and Empirical Researches in Urban Management
Volume 8 Issue 4 / November 2013

2.2. From Stakeholder Needs to Tools Development

One of the requirements for developing IT tools for urban management is the definition of system and/or software requirements in detail capturing multiple aspects or viewpoints. These detailed requirements enable software engineers to manage any conflict between these requirements using proven approaches, for example goal-driven techniques developed by van Lamsweerde et al (1998), before software is designed or implemented, resulting in reduced costs for software error correction and/or maintenance. Also, these requirements must be complete and consistent and clearly present both functional and non-functional aspects to ensure that software products meet user needs (Hanzl 2007). The real challenge in requirements development is that these requirements must be gathered from the relevant stakeholders or end users in order to be able to decide the correct functionality and usability of the IT tools to be developed. However, often end user or stakeholders are not accessible to provide and/or validate detailed requirements.

Furthermore, different stakeholders including urban planners, NGO's and the business community individually aim to secure the maximum benefit according to their own different requirements (Relhan et al, 2011). This suggests the need to structure stakeholders into specific categories that can help in prioritising and identifying more relevant requirements. Although where commonalties of requirements exist communication regarding this commonality can be inhibited if the domain vocabulary of contributing partners and the semantics of specific terms vary between different stakeholder groups. The expression "scenario" is a good example. Programmers understand scenario as a storyline to illustrate a use case, while in common parlance a scenario is identified as a defined, documented image of a possible future (Loibl & Walz 2010). Accordingly, in terms of urban development a scenario refers to a view of a future evolution of a certain location in the urban area. Clearly, it is essential to strive for a common understanding of the domain vocabulary and its semantics. In addition to the complexities identified above in the elaboration of just a few obvious examples of requirements specification, most users engaged in the city's planning process are not familiar with IT system development methodologies, processes and vocabulary, consequently there is a need for translation. User specifications have to be "translated" for the IT community (and vice versa) in a fully understandable form.

With these various considerations concerning requirements specification in mind, this paper aims to provide some insights into the kind of user requirements IT developers must consider and the translation issues arising.

2.3. Requirements Engineering Approaches

From a system and/or software development perspective several requirements engineering methodologies, processes, techniques and tools exist (Sommerville et al 1994; Loucopoulos and Karakostas 1995; Winter et al. 1995; Ould 1995; Holtzblatt and Beyer 1996; Kotonya and Sommerville 1998; Sommerville and Sawyer 1997; Nuseibeh and Easterbrook 2000; Chung et al 2000; Sutcliffe 2002; Maguire 1998). However, a generic requirement engineering process may consist of different activities: requirements elicitation, requirements analysis, requirements specification, requirements validation, quality management and requirements traceability, and requirements management.

In this regard Nuseibeh and Easterbrook (2000) provide a review of different techniques that can be utilised at different stages of a requirement engineering process. These techniques can include ethnography (Sommerville et al 1994), contextual enquiry (Holtzblatt and Beyer 1996), questionnaires, interviews, workshops, surveys, retrospection of documents, prototyping, goal-oriented approaches (Kavakli and Loucopoulos 2003; Kavakli 2002), process modelling (Ould 1995), scenarios (Sutcliffe 2003; Holbrook 1990; Whittle and Kruger 2004), paraphrasing models in natural languages, formal validation, various modelling techniques (Giaglis 2001) such as use-case modelling, behavioural modelling, contextual modelling etc. Our objective here is not to invent new methodologies and/or techniques, rather to select and apply best practice to gather user-defined requirements for urban management and share our experience with the aim of improving the process of the development of IT tools for urban management and governance.

3. CASE STUDY: THE URBANAPI PROJECT

In order to properly understand the CoReS method through its application to the UrbanAPI project (section 4), this section briefly introduces the project in order to become familiar with its background in the context of urban management and governance, the overall project development methodology and its three applications.

3.1. Background and Context of UrbanAPI Project

UrbanAPI aims to develop IT enabled tools supporting city governance and adapted governance models particularly addressing stakeholder engagement and citizen participation in the planning process, in order to enhance sustainable urban policy development and delivery.

These tools aim to provide planners and policy makers with the information they need to expose the socio-economic and environmental impacts associated with alternative options for territorial development, and at the same time create conditions in which the political mandate as a critical basis for more effective management, is secured (Yigitcanlar et al, 2008).

The main stakeholders addressed in the UrbanAPI project relate to three different interest groups: i) planning authorities interested in software solutions and application in their cities, ii) policy makers interested in content communication and means to intervene in the urban development process, and iii) ‘ordinary citizens’, laypersons with respect to methodology, but experts in local issue identification and specification of alternative development solutions.

In this regard urban planners from the project case study cities including Vienna (Austria), Bologna (Italy), Vitoria-Gasteiz (Spain) and Ruse (Bulgaria), are actively engaged in project related research, especially to define city policy related needs and system requirements.

3.2. UrbanAPI Tools Development Methodology and Applications

The UrbanAPI toolset aims to provide advanced IT-based intelligence in relation to three urban planning contexts and spatial scales as depicted in Figure 3. First, UrbanAPI directly addresses the issue of stakeholder engagement and citizen participation in the planning process by the development of enhanced 3D virtual reality (3DVR) visualisation of neighbourhood development proposals.

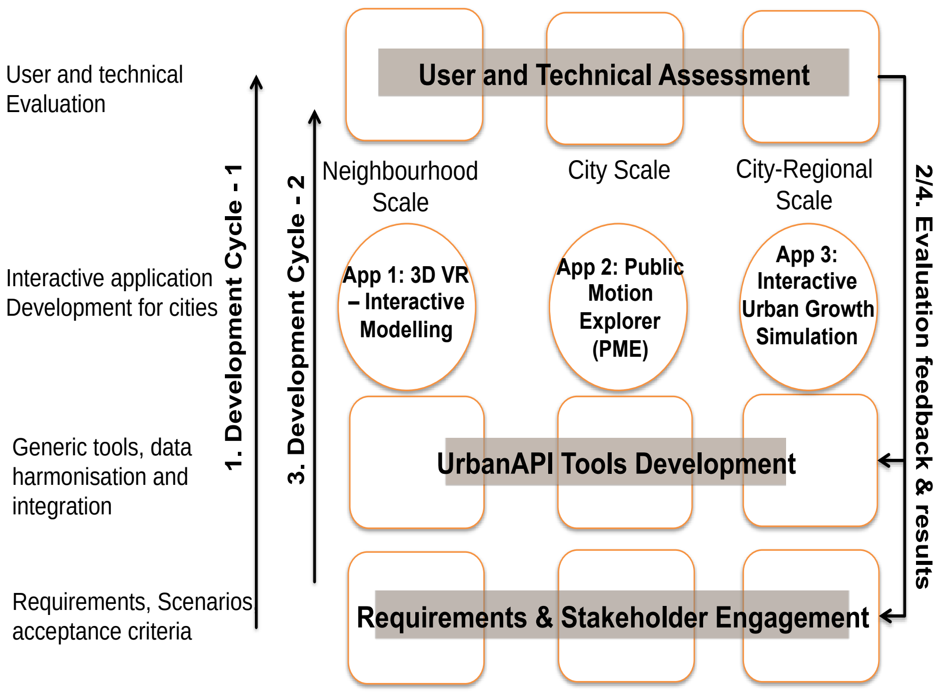
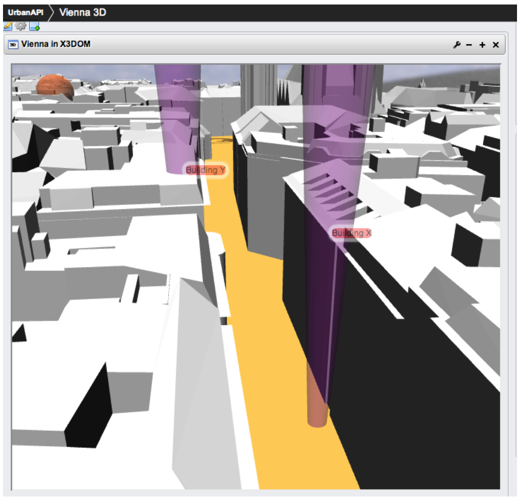
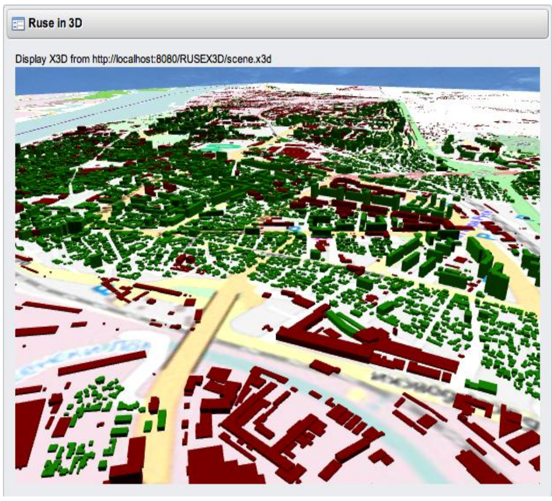


FIGURE 3 - URBANAPI: REQUIREMENTS, APPLICATIONS AND EVALUATION

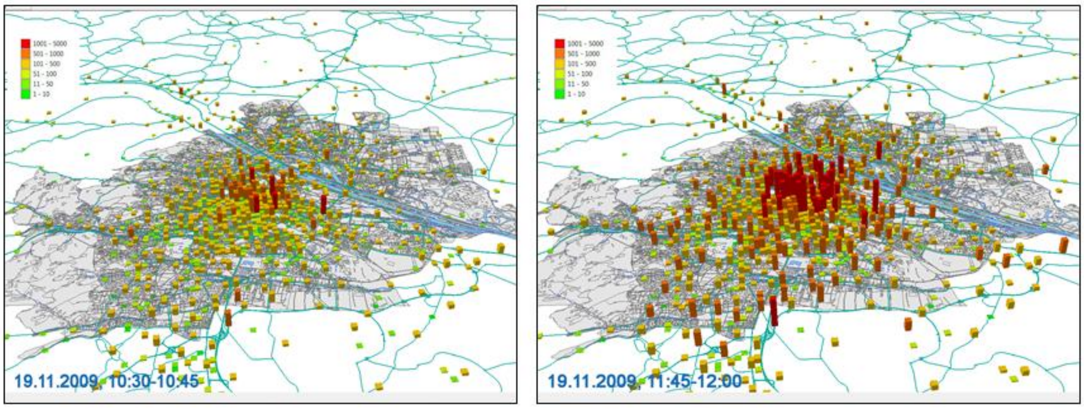
This 3DVR application also enables end users to access and propose amendments to planning proposals using an interactive visual interface via the web. Second, at the city-wide scale, UrbanAPI is developing a public motion explorer (PME) application, a mobile phone location based application using mobile phone location data (also known as GSM data), that permits the visual representation, and analysis of population distribution and movement patterns across the city, assisting planning agencies to explore space attractiveness and carry out mobility analysis.



a) Application 1: 3D VR application Vienna – adding annotations



b) Application 2: UGS application – 3D View of Ruse



c) Application 3: PME application: Population distribution dynamics in the Vienna Region during the morning hours (left: 10:30-10:45; right: 11:45-12:00) of a sample day 2009, (Source: Loibl & Peters-Anders, 2012)

FIGURE 4 -URBANAPI APPLICATIONS: A) APPLICATION 1: 3DVR APPLICATION VIENNA – ADDING ANNOTATIONS, B) APPLICATION 2: UGS APPLICATION – 3D VIEW OF RUSE, C) APPLICATION 3: PME APPLICATION – POPULATION DISTRIBUTION DYNAMICS IN THE VIENNA REGION (LEFT: 10:30-10:45; RIGHT: 11:45-12:00) OF A SAMPLE DAY 2009
Source: Loibl & Peters-Andres, 2012

Theoretical and Empirical Researches in Urban Management
Volume 8 Issue 4 / November 2013

Finally, UrbanAPI is developing urban growth simulation (UGS) for city-regions, addressing multiple urban planning challenges including visualisation of planning interventions and assessment of the impact of alternative proposals for urban expansion (and/or shrinkage) in the peri-urban area, and associated concerns for the optimum distribution of residential, employment and associated services.2 Basic snapshots of the visual interfaces of these three UrbanAPI applications are depicted in Figure 4.

UrbanAPI adopts the agile system development methodology, for example, SCRUM (Deemer et al 2012). This means multiple application development activities run concurrently and are repeated in multiple cycles (i.e. two cycles for UrbanAPI) in order to obtain improved results. Figure 3 shows that the overall process is initiated with 'Requirements gathering and stakeholder engagement followed by generic tools development which can be utilised for the development of 3DVR, PME and UGS applications for the participating cities.

The assessment of these applications assists in the identification of the limitations of these applications and/or the gathering of new requirements. Subsequently the entire development process is repeated. One of the major benefits of this agile methodology is regular stakeholder engagement at different stages of the toolset and application development process that assists in specifying and validating necessary and common requirements definition, resulting in generic and user defined IT tools.

3.3. UrbanAPI Application Matrix – City Scenarios

As indicated above the requirements for the development of the UrbanAPI IT toolset and applications are generated by the case study cities. Most of the cities are participating in two different applications that aim to explore the potentials of the applications in relation to context specific socio-economic, environmental and territorial characteristics, governance structures and practices, and that furthermore aim to define potential commonalities as a basis for the development of generic applications.

Table 1 identifies the city participation in the different UrbanAPI applications, defined according to local policy priorities.

TABLE 1 - URBANAPI CITY APPLICATIONS

	3D VR	Public Motion Explorer	Urban Growth Simulation
Vienna	√	√	
Bologna	√	√	
Vitoria-Gasteiz	√	√	
Ruse			√

[2] More detailed description of these three levels of applications can be found from http://www.urbanapi.eu

4. THE CORES METHODOLOGY FOR REQUIREMENTS DEVELOPMENT AND MANAGEMENT

In this section we present the CoReS methodology for requirements development in UrbanAPI. The CoReS method consists of the following five main components depicted in Figure 5 and performed in a specific order.

- Groundwork and Context
- Requirements Gathering Workshops
- Scenario-Based Requirements
- Requirements Specification Template and Validation
- Requirements Management

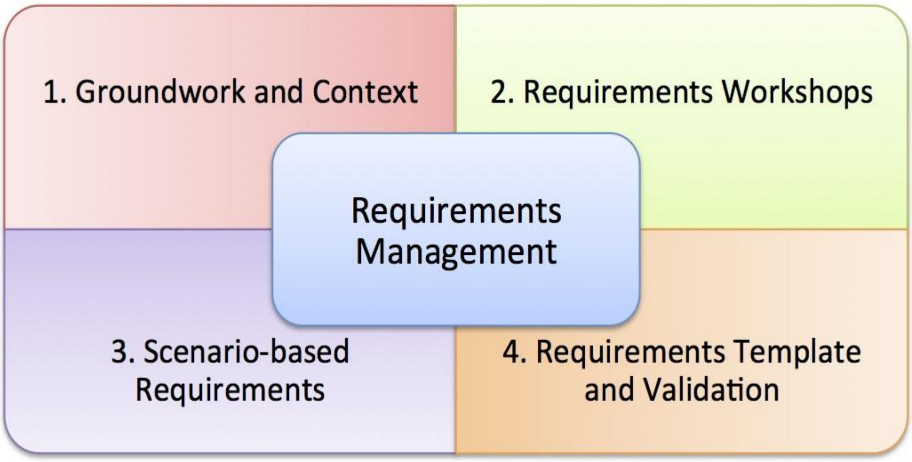


FIGURE 5 - MAIN COMPONENTS OF CORES METHOD

The overall process is coordinated and managed by a 'Requirements Expert' role. The five CoReS components are selected for the following prime reasons:

i) to understand the environment where solutions will be applied, with the objective to identify existing problems, stakeholders' needs, policy objectives and existing organisational processes;

ii) to understand stakeholders in order to develop user defined solutions by maintaining regular engagement with stakeholders throughout the requirements development process;

iii) to gather, specify and validate application requirements within specific timeframes without effecting other stages of the project development life cycle;

iv) to accommodate the needs and requirements of multiple stakeholders in various forms (i.e. scenarios, use cases, functional and non-functional requirements) with the objective to get better understanding of application requirements and to identify commonalities.

4.1. Groundwork and Context

The groundwork and context technique (Finkelstein 1993; Nuseibeh and Easterbrook 2000) is used to elicit basic requirements and establish an understanding of city needs, in terms of policy goals, and constraints in order to establish the basis for effective participation in the requirements gathering workshops. In UrbanAPI groundwork refers to the identification of stakeholder goals, conflicts, scope of the system and boundaries, associated risks and alternative scenarios.

Similarly, context refers to the rationale for development, for example the extent to which it is policy driven or market driven etc. In order to formally establish groundwork and context a questionnaire is prepared and distributed to city stakeholders (i.e. representatives from different departments of the city administration including urban and transport planning, survey, IT and GIS departments, public communication, environment and regional agencies, NGOs and policy makers) for their input before the requirements workshops.

Figure 6 depicts the overall approach to prepare the groundwork and establish the context for the development of requirements specifications for the UrbanAPI applications.

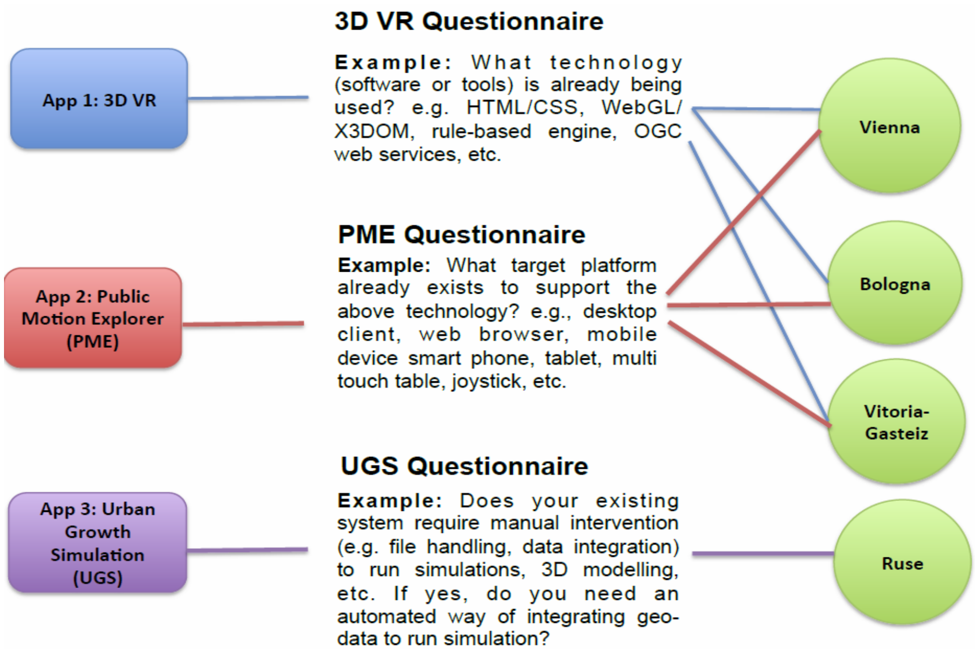


FIGURE 6 - URBANAPI: PREPARING GROUNDWORK AND ESTABLISHING CONTEXT

The questionnaire primarily addresses ‘what’ and ‘why’ type questions, divided into general categories according to specific project needs, and which assist in structured requirement gathering and analysis. The questions are structured in relation to the following widely applied categories:

Organisation Objectives

This category of questions aims to identify objectives, goals and high level requirements that are relevant to local needs and which can bring benefits to the municipality/organisation in various forms. The following are selected examples:

‘What are the city’s goals, in terms of priority, in developing this application? e.g. improving communication with citizens, addressing conflicts on planning choices, urban planning and policy development, etc.’

‘Can you identify any local use cases or scenarios where this application would be applied? e.g. city planning, public transportation such as Limited Traffic Zone, use and attractiveness of public space, and analysis of urban sprawl etc.’

Organisation Future Needs

This category of questions aims to indicate high level system requirements, functions, features, scope, interfaces, policy and standards compliance, challenges, expected outcomes and their measurability. The following are selected examples:

‘What are the current problems facing your organisation related to this application? And, what is required from this application to solve any of these problems?’

‘What specific functions would you like to see in the final product? e.g. data import/export function; user defined rules for modelling and simulations; automated workflow supporting data acquisition, processing and visualisation; use of specific algorithms, specific computational and visualisation models, etc.’

Existing System and Current Problems (if existing)

This category of questions aims to discover the limitations in the existing system, if they exist, and the required interface with the new applications. The following are selected examples:

‘What will the new application system accomplish that is not currently accomplished manually or with other systems?’

Do you use any special technical terms (vocabulary) in your environment? If so, which ones? And, in which form/language do these exist?’

Theoretical and Empirical Researches in Urban Management
Volume 8 Issue 4 / November 2013

Stakeholders and/or System Users

This category of questions aims to identify major stakeholders and their roles. The following are selected examples:

'Who is going to benefit most from the UrbanAPI applications?'

'Who are the main stakeholders for UrbanAPI tools and applications (and their expertise)?'

Acceptance Criteria

This category of questions aims to identify the measurable success indicators once the system has been developed. The following are selected examples:

'What is the expected application outcome e.g. a checklist including customer satisfaction, operational effectiveness, increased revenue, decreased cost, regulatory compliance? How will it be measured e.g. acceptance criteria?'

'From your perspective how would you define success for this particular application?'

Assumptions and Issues

This category of questions aims to identify any known issues and assumptions regarding the development of the system. The following are selected examples:

'Please indicate any unresolved issues which you consider can affect the application development? e.g. data availability for PME application, no 3D data, etc.'

'Are there any assumptions we have to make while developing the application? e.g. availability of latest data, transformations, dependency of other systems?'

4.2. Requirements Workshops

The requirements gathering workshops are used as an instrument for face-to-face communication between tool developers and city stakeholders. Figure 7 depicts the objectives and expected outcomes of the requirements gathering workshops. These objectives are to gather detailed requirements, based on direct discussion with the local city stakeholders, in order to acquire fuller understanding of local needs, policy issues and urban management goals. Furthermore, the identification of urbanAPI application specific scenarios for selected districts/regions, and associated data availability assist in defining the feasibility of application development for specific cities.

In UrbanAPI, prior to these workshops, introductory material for the three applications is distributed to city stakeholders to allow them to become familiar with UrbanAPI project. Participants to these workshop are mainly experts from different departments of city administrations including urban planning, transport planning, surveying, IT department and GIS experts, as well as regional and environment agencies, and in some cases local politicians (e.g. Deputy Mayor in Ruse).

Based on the groundwork and context component outcomes, the UrbanAPI applications are presented in specific city contexts. Then, city stakeholders present their local initiatives and projects where selected UrbanAPI applications can be applied.

This process leads to the identification of city preferences for using specific UrbanAPI applications and also creates the opportunity for discussions to clarify complex issues. For example, these discussions included identification of more specific user needs, restrictions on availability of mandatory data for applications, site identification to build scenario descriptions, and other expected outcomes from the UrbanAPI applications.

All these discussions are documented and verified by workshop participants providing the basis for the development of scenarios and detailed functional and non-functional requirements.

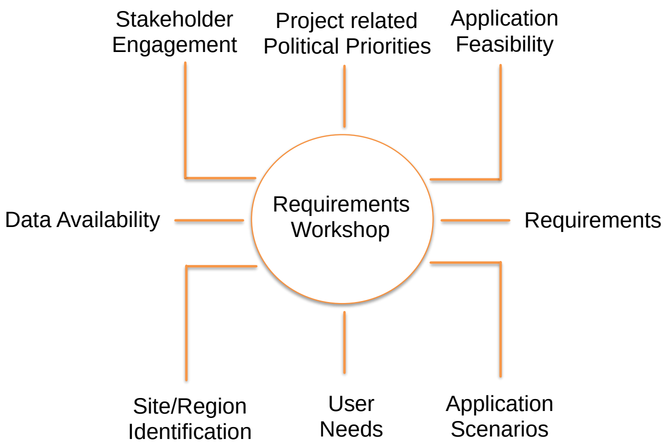


FIGURE 7 - REQUIREMENTS WORKSHOP OBJECTIVES AND EXPECTED OUTCOMES

4.3. Scenario-based Requirements

In order to secure a common requirements development approach and attract wider stakeholder participation, a scenario-based requirements and design technique has been used. In the literature, scenario based requirements elicitation, analysis and design approaches are commonly used (Kaindl 2005; Sutcliffe et al 1998; Sutcliffe 2003; Holbrook 1990; Whittle and Kruger 2004). In UrbanAPI, an operational scenario is referred as an instance of a task dealing with data handling (create, read,

Theoretical and Empirical Researches in Urban Management
Volume 8 Issue 4 / November 2013

update, delete operations), described in a user-specific language without including technical and/or implementation details. A scenario refers to a concrete narrative, or story, that describes the hypothetical use i.e. one future possibility for tool application. A scenario may articulate information including: who uses the system, what are the users roles and what are they trying to achieve – objectives and goals; what is needed in the application and why; the user-system interaction and the context in which the user will work with the system; the users constraints and limitations; clarifies what is regarded as a successful outcome of the application. This approach is useful to perform retrospective analysis and derive use cases to analyse and specify user requirements. Furthermore, it helps in presenting system specifications in user-specific terminology which can be more effectively validated by the end users including policy makers, planners, etc. Multiple scenarios are developed for each UrbanAPI application for each city. As an example, we present an excerpt from one scenario description. More detailed scenarios descriptions are published elsewhere (Requirements Definition – UrbanAPI, 2013).

4.4. Requirements Specification Template and Validation

In the CoReS method, a structured template is defined for requirements specification, and mainly used to elicit, analyse, specify and validate city requirements. For each city, it includes application based scenarios, user needs and goals, local stakeholders, use cases, functional and non-functional requirements. Glossary of these terms is provided in Annex – 1. Figure 8 uses UML notation and depicts the relationship (and multiplicities) between different entities for the development of requirement specification. It shows that each application of a city can have many scenario descriptions, user needs and goals and local stakeholders. For example, 3DVR is an instance of 'Application' entity that can have several 'user needs and goals' and 'local stakeholders'. Furthermore, many 'use-cases' can be derived from each 'scenario' description to clarify specific activities, and to define multiple 'functional' and 'non-functional requirements'. Such a structural composition for requirements specification provides flexibility in terms of capturing detailed application requirements with multiple viewpoints. Below we provide examples of these different entities.

Example excerpt of 3DVR application scenario for Vienna:

Scenario: Investigating architecture of future of city projects in architectural competitions

'Mr. Seidler works for the Urban Planning Department of the City of Vienna. A new urban development area will I be addressed . As usual an architects' competition has been held. As a result, Mr. Seidler gets a number of drafts (as digital 3D-data) from the architects. He would like to use an information system that can take architectural design data as an input and visualise 3D architectural model over city map and other socio-economic elements. Further, he would like to see visual effects and impact on the city if certain variables in architectural data are modified. This would help Mr. Seidler in comparing and contrasting various architectural design options for the development of architectural project and assessing its visual impact on the city. '

Description: Mr. Seidler loads 3DVR application using multi-touch table or he can connect his desktop/computer to the architecture and planning competition website using his web-browser. On the first screen he gets various options and text explaining what he can do. One of the options 'Architecture of Future' catches his attention. He selects this option and enters in to the system. A new screen is loaded and on the right side of the screen, a context sensitive help guides Mr. Seidler by showing the steps he needs to follow to visualise his architecture project. Following these steps, he clicks on the select data from the 'control panel' menu. Then he selects data from a data source (e.g. his computer) and clicks on the load data button and then a dialogue box shows that data loading is in progress. After a couple of seconds his architectural project data is loaded in the system. As soon as project data is loaded, he gets a hierarchical view of project data elements in the left side window whereas the visual representation of the architecture is shown in the major right side window (i.e. workspace). The help dialogue box shows that he is in editing mode. Then from the control panel he loads the city map and selects a specific area of the city where the new architectural construction would take place. Then in next step he clicks on the 'run project' from the control panel and a new window is loaded with 2D/3D map visualising the new architecture and its visual impact on the selected area (quarter/district). Mr. Seidler can choose to navigate around with an aerial view to visualise the new development impact. Mr. Seidler can also annotate different points on the visual screen such as possibility of new developments etc. He stops the 'run project' mode and goes back to 'editing' mode. In the editing mode, he changes some properties of the data fields (e.g. new/edited values such as height, colour, orientation etc) and uses different drawing tools to make visual changes in the architectural design and again runs the project from the control panel to see the visual impact. In editing mode he can also select another area on the map of the city. He can also choose measuring tool from the control panel to examine the height of a selected object and compare with neighbouring buildings e.g. how much higher the suggested high-rise is as compared to the church-tower next door. Having already selected the tool he can also measure the distance of the main entrance to the next tram stop. In the end he wants to make his project publically available either to a restricted circle of users for instance to discuss the project or proposed changes with the developers and politicians or he may want to share the project publically. This sharing of the project will allow others to make/recommend changes/modifications in order to

Theoretical and Empirical Researches in Urban Management
Volume 8 Issue 4 / November 2013

enhance the architecture and/or surroundings. He also wants to be notified for any such recommendations and/or modifications.

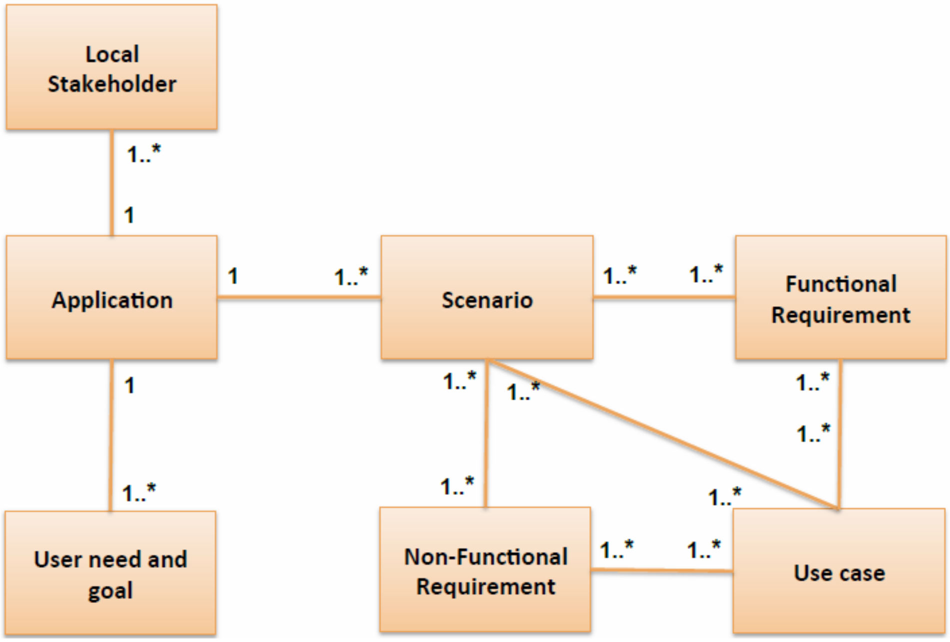


FIGURE 8 - REQUIREMENTS SPECIFICATION TEMPLATE ENTITIES

Example excerpt of user needs for 3DVR application in Vienna

The need to have 3D visualisation tools was realised about 10 years ago in the urban development projects near the Vienna centre in order to avoid negative visual impacts on the historic city centre, and so support the maintenance of the UNESCO world heritage status. Since then 3D visualisation has been one of the priorities for urban planning and as a result Vienna has a 3D database in CityGrid format. This data is used for different applications for example, a 3D visualisation system 'zacturn' developed by the 'z-koor company, with three prototypes installed at different places in the city for public use. Using this system, the entire city can be navigated to explore current and future possibilities in 3D. Currently five different urban development projects are accessible to the general public. These applications are functional, robust but not appropriate for engagement with a large number of public (i.e. over 2 million). Allowing remote web-based access and providing new possibilities not only for general public but also for expert views including urban planners and architects is needed. Vienna aims to support open access to data and open urban planning involving citizens. UrbanAPI 3D VR application can take one extra step to make 3D visualisation available to wider audience via web based access mechanisms in order to have wider public participation in planning issues with frequently updated 3D city data at low maintenance cost. Further, it will provide the opportunity to experts to visualise and evaluate the impact of a project on the city with efficient use/release of human resources of Surveying/IT department to perform other tasks.

Example excerpt of a selected Use Case Description for the above Scenario description in section 4.3

Use case UC_1.1_A: Stakeholder(s)/Architect(s) load city and architectural data into the system – **Essential**

Objectives and Goals: Enabling end users to load architectural design data and city data/map from CityGrid 3D DB in to the 3D VR system

Stakeholders: Architects and Urban Planners

Pre-conditions: Data is according to pre-defined formatting guidelines and common data model recommendations.

Post-conditions: A notification with text that data is successfully loaded in the 3D VR application.

Description: This use case aims to facilitate end user to load architectural data in the 3D VR application using web-browser and/or „multi-touch table“, a special large-scale interactive touchscreen device.

Following sequence of steps can be followed:

i) Start 3D VR application using a „multi-touch table“ or connecting to 3D VR website using a specific web address.

ii) Select „Architecture of future“ option and click next

iii) Select „load architectural data“ option from control panel

iv) Select data source and click „load data“

v) Data is loaded in to the application and displayed in both a hierarchical view and respective visual architectural design

vi) Load city data from Vienna CityGrid 3D database and associate the architectural data with specific geo-coordinates.

Each functional and non-functional requirement is further detailed to acquire common understanding of the requirement statements, rationale, known assumptions, acceptance criteria and validation by the requirement owners. Furthermore, each requirement statement follows the MoSoCow method (2012) to

Theoretical and Empirical Researches in Urban Management
Volume 8 Issue 4 / November 2013

indicate the importance of implementing a solution to the requirement. The following table (Table 2) shows the required attributes for each requirement statement, e.g. a functional requirement statement can be:

'System must be able to show population density in census cells during work days and peak morning hours between 7:30 am and 9:30 am.'

and associated with the following attributes, indicated in Table 2 (and an example shown in Table 3).

TABLE 2 - REQUIREMENT DEFINITION ATTRIBUTES

Attribute	Possible Values
Scenario ID	*Refers the Scenario from which this requirement is initiated*
Use case ID	*Refers to the associated use case(s)*
Requirement ID	*Unique identifier for traceability purposes e.g. FR[City number-Application number-Requirement number]. For example FR1.1.1 means city 1 (Vienna), Application 1 (3D VR) and Requirement number 1. Default City order is: Vienna 1, Bologna 2, Vitoria-Gasteiz 3, and Ruse 4. Default application order is: 3DVR 1, PME 2, and UGS 3.*
Owner/Responsible	*Initiator and in charge of this requirement who will validate it / Contact person*
Rationale	*Justification for the requirement*
Created By	*Author of the requirement*
Stakeholder	*Potential stakeholders effected by this requirement*
Function	*Statement of the main functionality*
Version	*Version number starting from 1*
Assumption	*Any assumptions related to this requirement which can effect design and implementation stages*
Status	*Any one of these values {Original / Changed / Complete / In complete / Dropped}*
Priority	*Any one of these values {Essential / Desirable / Optional}*
Traceability	*Backward and Forward traceability (new requirement identification numbers)*
Requirement Change	*Reason for change in requirement and changed requirement ID*
Approval status	*The value of this attribute is set after analysing the technical feasibility of the requirement. Any one of these values supported by a justification statement should be provided {Initiated / Approved / Not-Approved}*
Means of validation	*Possibilities may include {Testing / Validation / Demonstration / Survey etc}*
Requirement Validated by Owner/Responsible	*This attribute is used to get possible values {Yes /No /With changes}*
Acceptance Criteria	*Indicators and standards that satisfy the Owner/Responsible persons' quality and functionality expectations for this requirement, and gain acceptance of the final product.*

Example excerpt of a Functional Requirement Definition for Vienna 3DVR application

FR1.1.3: Common data models, data transformation and solutions should be provided to architects and planners to prepare for, and participate in architectural competitions.

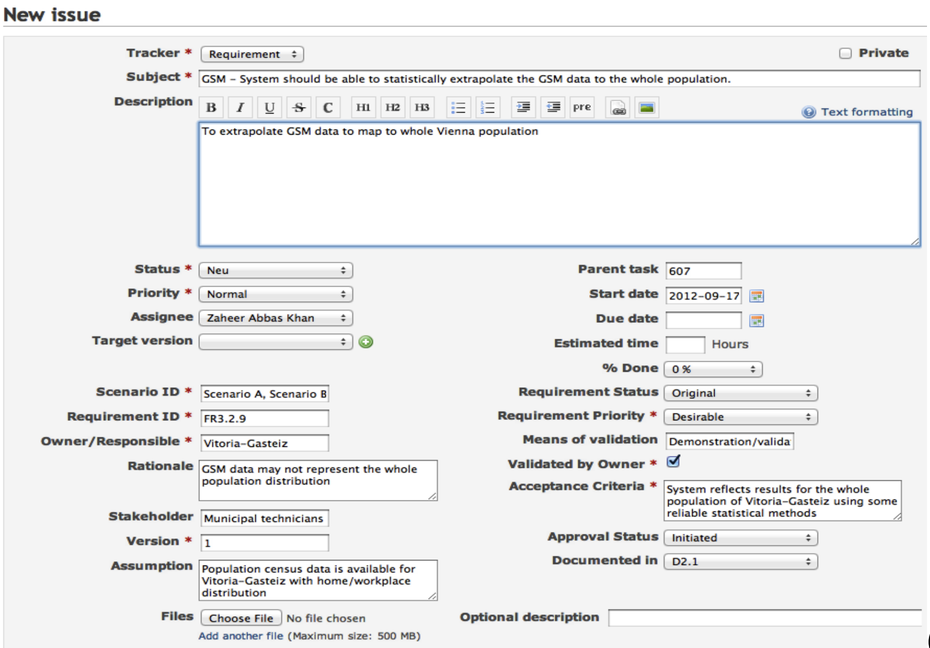
TABLE 3 - AN EXAMPLE OF A FUNCTIONAL REQUIREMENT ATTRIBUTES

Attribute	Value	Attribute	Value
Scenario ID	*Scenario A and B*	*Use case ID*	*UC_1.1_A*
Requirement ID	*FR1.1.3*	*Owner/Responsible*	*City of Vienna/[Names]*
Rationale	*Architectures use various different tools and data models which may not be compliant and used in a common system to visualise and compare results*	*Created By*	*[Names]*
Stakeholder	*Architects, Planners, Public*	*Function*	*Providing documentation of common data model, guidance document for harmonisation and scripts/tools to perform harmonisation of the architecture data that can be used in 3D VR application.*
Version	*1*	*Assumption*	*There is no general/common data model for the application*
Status	*Original*	*Priority*	*Essential*
Traceability	*Nil*	*Requirement Change*	*N/A*
Approval status	*Initiated*	*Means of validation*	*Demonstration, WP5 Evaluation*
Requirement Validated by Owner/Responsible [Yes/No/With changes]	*Yes*		
Acceptance Criteria	*Successful use of Architecture data from various source applications (e.g. Studio 3D Max, AutoCAD, Cinema 4D etc.) in 3D VR application*		

At this stage of the CoReS method, the 'Requirements Expert' plays a crucial role by acting in liaison between IT developers and city stakeholders. On the one hand this helps in gaining understanding about the needs of IT tools from the user perspective as well as the expectations of system modellers and IT developers, to clearly specify what is needed. On the other hand the requirement expert plays a critical role in requirements negotiation, and detecting preliminary conflicts in requirements e.g. scope of the application features i.e. spatial coverage, security concerns e.g. use of external web services etc, and suggested resolution to have a more feasible and validated requirements specification. In urbanAPI the requirements specifications are refined based on the requirements negotiation between system modellers, tools developers and city stakeholders coordinated by the UrbanAPI requirements expert. A continuous review technique is applied, in which every new version of the requirements specification is reviewed by the requirements expert to detect any conflicts in requirements. If conflicts are detected then negotiation activity is initiated between the relevant stakeholders with the objective to achieve a revised requirement definition that is acceptable to all.

4.5. REQUIREMENTS MANAGEMENT AND DEVELOPMENT PROCESS

All requirements collected help in establishing a common understanding between the IT developers and the user communities concerning the IT tools development. UrbanAPI project development is based on the agile development methodology, which involves mechanisms for the controlled evolution of requirements in response to various factors including for example: hidden requirements, new requirements, changes in environmental and/or political priorities, and requirements contradictions revealed at design or implementation stages.



(a)

(b)

FIGURE 9 - REQUIREMENTS ATTRIBUTES IN REDMINE MANAGEMENT TOOL: A) REQUIREMENTS ENTRY, B) REQUIREMENTS VIEW FOR CHANGE MANAGEMENT

In the above context UrbanAPI's project management e-infrastructure uses the Redmine management tool (2012) in order to manage the requirements changes. All requirements are inserted in the Redmine database and each requirement is issued a ticket/issue number to ensure the traceability of the requirement from design to implementation (software features) and also to facilitate testing and evaluation. In particular the Redmine tool facilitates the management of requirements changes arising from the evaluation of cycle 1 of the agile methodology. Figure 9 depicts the requirements definition attributes (Tables 2 and 3 above) as implemented in the Redmine management tool.

In order to apply the CoReS method more effectively, a process-based approach has been used in UrbanAPI, as shown in Figure 10. Primarily, this process ensures that CoReS components are instantiated and managed in a specific order resulting in a specific outcome artefact i.e. questionnaire, requirements, application scenario, detailed and validated requirements. This process may repeat selected components if, for some reason, the stakeholders require changes in requirements before validating requirements. The iterative phase aims to improve the requirements specifications in an attempt to secure validated requirements and meet acceptance criteria. As a result, the final output of the development process is a detailed requirements specification that includes validated scenarios, use cases, user needs and goals, functional and non-functional requirements for each city-specific application.

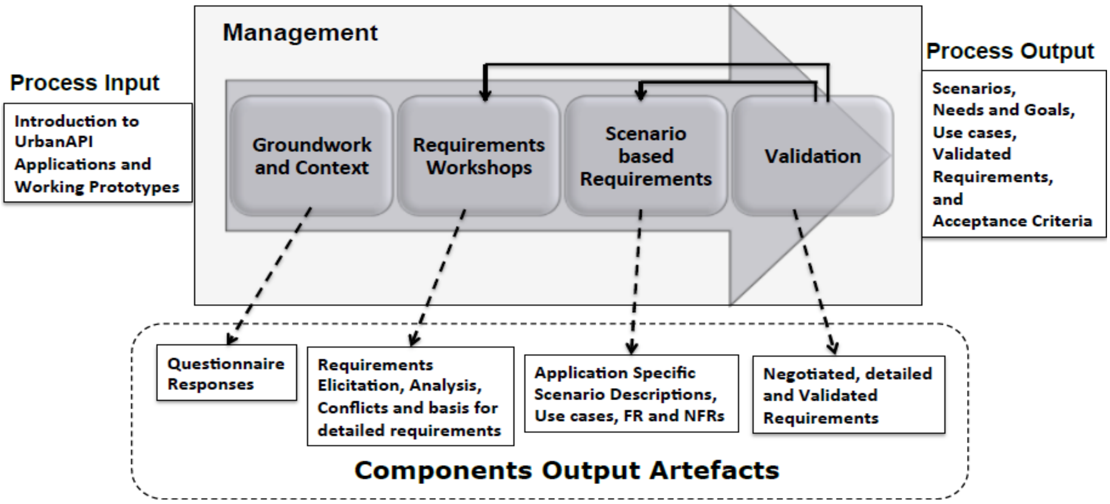


FIGURE 10 - CORES DEVELOPMENT PROCESS

4.6. Identifying Potential Stakeholders

Stakeholders play an important role in the development of a project especially deriving and specifying requirements for the development of tools and applications, data provision and validating the outcomes of the project. IT tools for urban management and governance may fail if relevant stakeholders are not identified and engaged in the requirements development process to secure complete or sufficient system requirements. Typically, stakeholders include individuals or organisations that have a vested interest (i.e. financial, technical, knowledge, policy etc.) in the outcomes and/or success of the project. Stakeholders can be individuals who are involved in the development of the system (i.e. operators, developers or maintenance staff) or use it (i.e. functional or operational beneficiaries) or who are mainly interested in the outcome of the project (political or financial beneficiaries).

Different techniques can be used to identify and classify stakeholders, e.g. Primary, Secondary and Tertiary stakeholders (Sutcliffe 2002). In order to identify and structure UrbanAPI stakeholders, the 'Onion Model' technique is applied that indicates political beneficiaries, functional beneficiaries, normal users or operators and support or maintenance operators (Alexander 2005).

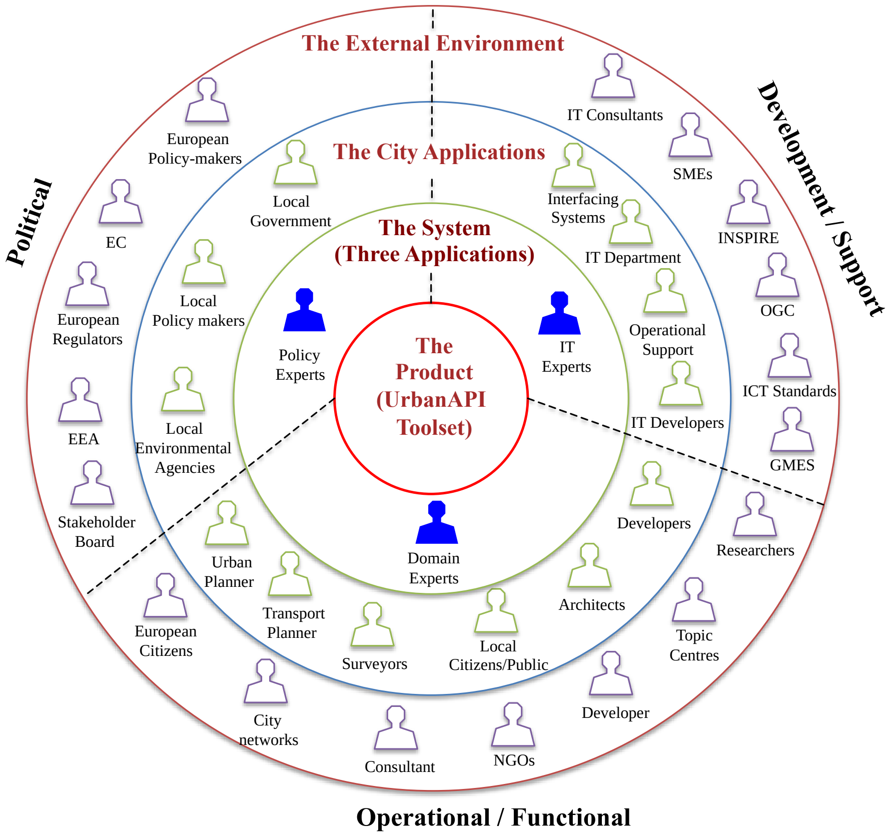


FIGURE 11 - URBANAPI STAKEHOLDERS BASED ON THE ONION MODEL

According to the onion model Figure 11 depicts the detailed relationship of possible UrbanAPI stakeholders (i.e. 'actors' as 'roles') which consists of four concentric layers each of which denotes a subset of stakeholder entities relevant to the development of the project:

a) 'The Product' is the inner layer referring to the UrbanAPI Toolset that provides basic functionality for the development of the UrbanAPI applications.

b) 'The System' refers to the three applications of UrbanAPI that includes the UrbanAPI Toolset and its human operators, local policy or domain specific regulations and standard operating procedures, etc.

c) 'The City Applications' refer to 'The System' plus any human beneficiaries e.g. local city stakeholders.

d) 'The External Environment' refers to 'The City Applications' and any other stakeholders who are not directly involved in the development of the project e.g. regulators, decision makers, Stakeholder Board, etc.

5. CRITICAL ASSESSMENT, DISCUSSIONS AND REFLECTION

In an attempt to respond to the basic research question identified in section 1, this section aims to critically assess the strengths, and weakness of the CoReS method, and reflects on its usefulness for the urban management domain in general and the UrbanAPI project in particular.

In general, the CoReS method facilitates the collection of requirements at different levels of abstraction i.e. scenarios, use-cases and specific requirements definition which represent the requirements of different types of stakeholders, critical in the design and implementation and subsequent validation and acceptance of the IT tools. However, with respect to usefulness of the CoReS method, one has to consider different perspectives and the roles of stakeholders involved in the development of IT tools for an urban system. For instance, the urban planner's viewpoint on the overall CoReS method, in general, and system requirements, in particular, is unlikely to be identical to the application analyst's or modeller's or IT expert's point of view.

In this context, the authors, contributing to this paper, represent different roles in the application of the CoReS method to the UrbanAPI project and the overall development of UrbanAPI toolset and applications for cities including: i) IT or software engineer, ii) land-use planner, and iii) application scientist or modeller. Accordingly the authors are able to compare the different requirements specifications from each viewpoint as presented in the following sub-sections.

5.1. Technical perspective on CoReS method

From an IT perspective, the CoReS method provides numerous ways to develop a basic understanding of urban management issues and gather detailed system requirements as demonstrated through the UrbanAPI case study. In addition, using a software tool for requirements management (e.g. Redmine system) helps in storing necessary and valuable information to better plan the implementation of system features and manage requirements changes and identify the impact on other related system tasks.

Performing such a detailed requirements development process can be time consuming and requires sufficient resources, but the returns on these investments are potentially substantial. For example, the requirement definition template (section 4.4) facilitates specification of requirement description at both abstract and detailed levels which provides good understanding of rationale, traceability, assumptions and acceptance criteria that helps in requirements analysis and performance of feasibility studies. In addition, software developers get a good idea about the expected workflow of the application functions and acceptance criteria against which the final products will be validated. Such a detailed requirements specification also helps in analysing and identifying common requirements from which to develop generic software features which can be applied in other cities. In addition, these commonalities help in avoiding the reinvention of the wheel (i.e. developing something which has already been implemented).

For instance, in UrbanAPI the following tables (Tables 4 and 5) represent the total number of scenarios, use cases and requirements defined for the 3DVR and public motion explorer applications for each case study city.

TABLE 4 - 3D VR APPLICATION STATISTICS (* THERE ARE COMMON USE CASES BETWEEN DIFFERENT SCENARIOS OF THE CITY APPLICATION)

Application	**3D VR Application Statistics**		
Cities	**Number of Scenarios**	**Number of Use cases**	**Number of Requirements**
Vienna	2	14	22
Bologna	3	15*	18
Vitoria-Gasteiz	3	11*	18

Preliminary analysis derived from the UrbanAPI CoReS method application indicates that nearly 50% commonality exists between two or more city requirements for the 3DVR application. These common requirements include usability aspects such as the need for visual aid; data synchronisation and integrity; public participation and ease of interaction with the application; accessibility of the application using different platforms e.g. web, smart phones; change impact assessment; importing new data and exporting results in common formats; and conformance to city administration IT policies.

This suggests, based on the commonalities in city requirements, that generic tools and services can be developed which can be exploited in fulfilling the specific needs of a wide range of city administrations in Europe. Furthermore, individually specified requirements from cities can also indicate new potentials which can be developed to enhance functional features and improve the overall range of features of the 3DVR application.

TABLE 5 - PUBLIC MOTION EXPLORER APPLICATION STATISTICS (* THERE ARE COMMON USE CASES BETWEEN DIFFERENT SCENARIOS OF THE CITY APPLICATION)

Application Cities	Public Motion Explorer Applications Statistics		
	Number of Scenarios	Number of Use cases	Number of Requirements
Vienna	3	8	22
Bologna	2	7	13
Vitoria-Gasteiz	3	20*	18

In a similar fashion to the 3DVR application, there exist about 60% commonalities between two or more city requirements for the public motion explorer (PME) application. These requirements include usability aspects such as need for intuitive user interface; visualisation of aggregated population; indication of places attractive to the public; extrapolation of results to the overall city population; identification of social biasing; visualisation of motion traces between city districts/zones; intra-city and extra-city origin-destination matrices and travelling mode; accessibility using different platforms; importing new mobile phone data and exporting results in common formats; integration of PME data with data from other sources e.g. GPS, survey information, city administration IT standards compliance; and workflow documentation.

The above analysis for PME strengthens the claim in the research question that generic tools can be developed to facilitate wider adoptability of the PME application where more specific requirements from cities add value to these common capabilities.

5.2. The urban management perspective on CoReS method

The CoReS method provides a critical framework in which questions concerning the current requirements of urban planning and policy making systems can be effectively analysed. The prototype solutions of 3DVR, PME and UGS applications play a major role in providing contextual understanding of the IT applications, demonstrating to stakeholders the variety of potential benefits offered by these IT applications as urban management and governance tools. Furthermore, as a user driven process involving all key urban and IT stakeholders, this process not only provides a rigorous examination of problems and development opportunities, but also creates a new dynamic in the mix of stakeholder participants. This dynamic of engagement between stakeholders as part of the requirements gathering process introduces novel perspectives that can generate new opportunities for the specification of the urban planning system requirements, and therefore offers a stimulus to consider new IT-enabled solutions, particularly in the context of the examination of potential commonalities in application specification.

New IT enabled solutions are required, as there is a need to respond to existing transformational governance demands including, for example, the need for effective communication between all stakeholders involved in the urban management process. But there is also need to consider future development requirements and the evolution of urban management systems over time. One of the key strengths of the CoReS method is it's ability to combine a rigorous analytical process with the full engagement of all relevant stakeholders, supporting the development of short-term solutions, and offering the potential to articulate longer term visions essential to the definition of future perspectives for IT enabled urban management and governance. The undoubted strengths of the CoReS based requirements gathering process lie in it's ability to create an effective and structured framework of engagement between key stakeholders in defining both specific development solutions as well as broader visions for the medium term evolution of the system.

The urban management perspective addressed by scenarios, use cases and functional and non-functional requirements specification is primarily focused on the functional and data related issues of case study cities. In fact, CoReS scenario descriptions are mainly related to use of the application tools but these scenario descriptions do not explicitly indicate how these tools fit in the overall policy making or urban management and governance processes; which activities of these processes are supported by these tools; and how data flow and transformation takes place within these processes. This suggests that a process-oriented approach (Khan Z, et al 2011) can also be introduced in the CoReS method by using process modelling languages such as RAD or BPMN to elaborate city administration governance and public participation processes with the objective to improve and transform these processes with the help of innovative IT tools.

5.3. The urban application modeller or analyst's perspective on CoReS method

The urban system complexity requires conceptual application specific modelling and analysis to understand cause-effect relations of the interactions between different environmental and socio-economic factors, and simulation of their overall impacts. Requirements specification for such application modelling and analysis aspects are most closely linked to the conceptual system development-side of the software tools e.g. statistical modelling. For instance, the UrbanAPI UGS application relies heavily on such statistical behavioural modelling to perform ex-ante simulation of urban growth development of the Ruse city-region. Often these requirements are not addressed either by representatives of the city stakeholders as their main focus is on solutions for urban planning problems. The assumption is that IT developers concentrate on requirements gathering more from data and programming perspectives and often miss essential modelling aspects. Therefore IT tools are either unable to fully implement the required functionality or/and often lack necessary features required for exploring urban systems and their interdependencies.

The CoReS method, as demonstrated in the UrbanAPI project, provides mechanisms to ensure that a regular dialogue is maintained between modellers, analyst and other stakeholders to develop complete, consistent and correct requirements specification. However, the requirements definition attributes (Table 2) refer primarily to the meta-information concerning software requirements and development progress, and do not monitor system model development progress and its related quality aspects. The reason for this is that quality and progress indicators are hard to be identified and that they cannot be defined in advance as the modelling process for urban aspects is, to some extent, a sequential process where model results refer to equations and model layout, and cannot be predicted in advance. This suggest that at the requirements development stage of a system development life cycle, consideration of additional aspects for requirements definition can improve the quality of requirements as well as extend the requirements specification, for example, data description, model functions description, expected results description from a modellers view. Such quality criteria could be integrated, for example as narrative logbook entries to allow a forward-backward tracing of system modeling decisions. Such an extension in the CoReS method would increase the usefulness of this method and would more effectively support the systems model developers' perspective in order to improve the conceptualisation and operationalisation process for any particular urban model.

In general, the above discussion and the application of CoReS method in the UrbanAPI project reveals that requirement development tools and techniques can be applied to the urban domain with some scope for improvement. With respect to the research question, we can conclude that the CoReS method is a generic requirements development process that can, to some extent, support the development of common user defined requirements specification for the development of generic IT-enabled urban management and governance tools which can be applied in many cities.

6. CONCLUSIONS

This paper demonstrates that the CoReS method provides a useful way to gather and analyse requirements and engage with planners, modellers and software developers. Furthermore, it gathers

Theoretical and Empirical Researches in Urban Management
Volume 8 Issue 4 / November 2013

requirements at different levels of abstraction i.e. scenarios, use case descriptions and requirement definitions that helps in capturing the perspectives of different stakeholders involved in the development of the urban management system. Furthermore, use of software tools (e.g. Redmine) helps in managing evolving requirements. These requirements may be extended to fulfill the needs of policy makers, urban system analysts as well as citizens, allowing sufficient involvement of these participant groups, and ensuring that appropriate IT tools are developed facilitating acceptance by a wide range of cities.

Based on the CoReS method application in the UrbanAPI project, a significant percentage of common application requirements generated by case study cities was identified which can provide the basis for the development of generic functionalities in the IT tools. For example, a preliminary analysis of the UrbanAPI requirements specification for four different European cities indicates that there are over 50% commonalities in policy development and decision making processes, supporting the design and implemention of generic capabilities in the interactive IT enabled participatory applications at different governance scales applicable to a wide range of European cities.

However, many requirements for the development of the urban management system vary between cities, which may be attributed to the local specifics including policy needs and priorities. Furthermore, the quality of data also varies from one city to another, placing constraints on the development of common solutions for different cities and so challenging the IT community to adopt software design patterns which permit the implementation and integration of these more specific features with the generic software systems. Based on the above findings, the CoReS method will be improved with new requirements gathering and analysis techniques e.g. process based approach and system modellers' perspective as a future research direction.

Acknowledgement

Research presented here was partly carried out within the project “urbanAPI” (Interactive Analysis, Simulation and Visualisation Tools for Urban Agile Policy Implementation), funded from the 7th Framework Program of the European Commission, call identifier: FP7-ICT-2011-7, under the grant agreement no: 288577, started in October 2011. We acknowledge the support provided by different partners of the urbanAPI project and departmental colleagues at different stages of the requirements development stages. In particular, our special thanks to Dr. Stewart Green of Faculty of Environment and Technology, University of the West of England, Bristol, UK for providing valuable inputs to overall CoReS requirements process and improving this manuscript.

REFERENCES

Alexander I. (2005). A taxonomy of stakeholders: Human Roles in System Development, *International Journal of Technology and Human Interaction*, 1(1), 23-59.

Boyd, S., & Chan, R. (2002). Placemaking: tools for community action. CONCERN, Inc. & Environmental Simulation Center, Denver, USA. Retrieved October 22, 2013, from http://www.sustainable.org/images/stories/pdf/Placemaking_v1.pdf

Chung L., Nixon B., Yu E. & Mylopoulos J. (2000). *Non-Functional Requirements in Software Engineering*, Boston: Kluwer Academic Publishers.

Cysneiros M., Leite J. & Neto J. (2001). A Framework for Integrating Non-Functional Requirements into Conceptual Models, *Requirements Engineering*, 6, 97-115.

Dahl, R. (1998). *On Democracy*. New Haven, Yale University.

Davies R. S., Selin C., Gano G. & Pereira G. Â. (2012). Citizen engagement and urban change: Three case studies of material deliberation, *Cities Elsevier Journal*, 29, 351-357.

Deemer P., Benefield G., Larman C., & Vodde B. (2012, December 17). The SCRUM Primer: A lightweight guide to the theory and practice of Scrum Version 2.0, Retrieved July 4, 2013, from http://www.infoq.com/minibooks/Scrum_Primer.

Eckhardt F. & Elander I. (2011). *Urban Governance in Europe*, (Eds) p360, ISBN: 978-3-8305-1502-9, BWV -VERLAG.

Felt, U. & Wynne B. (2007). Taking European Knowledge Society seriously. Report of the Expert Group on Science and Governance to the Science, Economy and Society Directorate, Directorate-General for Research, European Commission. Sixth Framework Programme, EC.

Finkelstein A. (1993). *Requirements Engineering: an overview*, In proceedings of the 2nd Asia-Pacific Software Engineering Conference, Tokyo, Japan.

Giaglis G. (2001). A Taxonomy of Business Process Modelling and Information Systems Modelling Techniques, *The International Journal of Flexible Manufacturing Systems*, 13(2), 209 – 228.

Hanzl M. (2007). Information technology as a tool for public participation in urban planning: a review of experiments and potentials, *Design Studies*, 28(3), 289-307.

Held, D. (2006). *Models of Democracy*, Stanford University Press.

Henderson P. (2006). Why large it projects fail. *ACM Trans. Program. Lang. Syst*, 15(5), 795–825.

Holbrook, H. (1990). A scenario-based methodology for conduction requirements elicitation, *ACM SIGSOFT Software Engineering Notes*, 15(1), 95-104.

Holtzblatt K. & Beyer H. (1996). *Contextual Design: Principles and Practice*, In book - Field methods casebook for software design, ISBN:0-471-14967-5, (pp. 301-333), New York: John Wiley & Sons, Inc.

IBM Requirements Management, (2009, December), Reducing rework through effective requirements management, IBM White paper, Retrieved October 22, 2013, from ftp://ftp.software.ibm.com/software/sg/rational/podcasts/2011/RAW14192USEN.PDF

Kaindl, H. (2005). A scenario-based approach for requirements engineering: Experience in a telecommunication software development project. *System Engineering*, doi: 10.1002/sys.20030, 8: 197–210.

Kavakli E. (2002). Goal-Oriented Requirements Engineering: A Unifying Framework, *Requirements Engineering Journal*, Springer-Verlag London, 6(4), 237-251.

Kavakli, E. & Loucopoulos P. (2003). *Goal Driven Requirements Engineering: Evaluation of Current Methods*, the 8th CAiSE/IFIP International Workshop on Evaluation of Modeling Methods in Systems Analysis and Design (EMMSAD '03), Velden, Austria.

Kearns, A. & Paddison, R. (2000). New challenges for Urban Governance, *Urban Studies*, 37(5-6), 845-850.

Khan, Z., Odeh, M. & McClatchey, R. (2011). Bridging the gap between business process models and service oriented architectures with reference to grid environment, *Int. J. Grid and Utility Computing,* 2(4), 253-283.

Khan, Z., Ludlow D., McClatchey R. & Anjum A. (2012). An architecture for integrated intelligence in urban management using cloud computing, *Journal of Cloud Computing: Advances, Systems and Applications*, doi:10.1186/2192-113X-1-1, 1:1.

Kingston, R. (2007). Public Participation in Local Policy Decision-making: The Role of Web-based Mapping, *The Cartographic Journal, ICA Special Issue*, 44(2), 138-144.

Kotonya, G. & Sommerville, I. (1998). *Requirements Engineering: Processes and Techniques*, John Wiley & Sons Ltd.

Kraemer, M., Ludlow, D. & Khan, Z. (2013). *Domain-specific languages for agile urban policy modelling*, 27th European Conference on Modelling and Simulation, 27-30 May 2013, Ålesund, Norway.

Liberatore, A. & Funtowicz, S. (2003). Democratising' expertise, 'expertising' democracy: what does this mean, and why bother? *Science and Public Policy*, 30(3), 146-150.

Loibl,W. & Peters-Anders, J. (2012). *Mobile phone data as source to discover spatial activity and motion patterns*, GI-Forum 2012, Salzburg, Austria.

Loibl, W. & Walz, A. (2010). Generic Regional Development Strategies from Local Stakeholders' Scenarios - an Alpine Village Experience, *Ecology and Society*, 15(3), 3.

Loucopoulos, P. & Karakostas, V. (1995). *System requirements engineering*, ISBN: 0077078438, McGraw-Hill, NY.

Maguire, C. M. (1998, July 16). *User-Centred Requirements Handbook*, RESPECT Consortium (ed.), Version 3.3. Retrieved September 24, 2012, from

http://www.idemployee.id.tue.nl/g.w.m.rauterberg/lecturenotes/UserCenteredRequirementsHandbook.pdf

Misuraca, G., Reid, A. & Deakin, M. (2011). Exploring emerging IT-enabled governance models in European cities - Analysis of the Mapping Survey to identify the key city governance policy areas most impacted by ICTs, Report by European Commission JRC-IPTS. Retrieved June 25, 2013, from ftp://ftp.jrc.es/pub/EURdoc/JRC65581_TN.pdf.

MoSCoW Method, (2012), Retrieved July 1, 2013, from http://en.wikipedia.org/wiki/MoSCoW_Method.

Montgomery, M. R., Stren, R., Cohen, B. & Reed, H. E. (2003), *Cities Transformed: Demographic Change and Its Implications in the Developing World,* (Eds), The National Academies Press, Washington DC.

Nuseibeh, B. & Easterbrook, S. (2000). *Requirements Engineering: A Roadmap*, ACM New York, NY, USA

Ould, M. (1995). *Business Processes: Modelling and Analysis for Re-engineering and Improvement*, ISBN # 0-471-95352-0, John Willey and Sons.

Pierre, J. & Peters, G.B. (2000). *Governance, Politics and the State*, Houndmills, Macmillan Press.

Poplin, A. (2012). Playful public participation in urban planning: A case study for online serious games, *Computers, Environment and Urban Systems*, ISSN 0198-9715, 10.1016/j.compenvurbsys.2011.10.003, 36(3), 195-206.

Redmine, (2012), Retrieved July 1, 2013, from http://www.redmine.org/.

Relhan, G., Kremena, J. & Rumana, H. (2011). Good Urban Governance through IT: Issues, Analysis and Strategies, The World Bank – A Knowledge Product. Retrieved July 1, 2013, from http://siteresources.worldbank.org/INTAFRICA/Resources/IT_Urban_Governance_Final_pub.pdf

Requirements Definition - UrbanAPI, (2013), Project Deliverable D2.1: Requirements Definition, Retrieved October 22, 2013, from http://www.urbanapi.eu/.

Rhodes, R. A. W. (1997). *Understanding Governance. Policy Networks, Governance, Reflexivity and Accountability*, Buckingham, Open University Press.

Rowe, G. & Fewer, L. (2005). A Typology of Public Engagement Mechanisms, *Science, Technology, & Human Values,* 30(2), 251-290.

Ruble, A. B., Stren, E. R., Tulchin S. J. & Varat H. D. (2001), *Urban Governance Around the World,* (Eds), Comparative Urban Studies Project, Retrieved October 11, 2012, from http://www.wilsoncenter.org/sites/default/files/urbangov.pdf.

Sommerville, I. & Sawyer, P. (1997), *Requirements Engineering: A Good Practice Guide*, John Wiley & Sons Ltd.

Sommerville, I. Bentley, R., Rodden, T. & Sawyer, P. (1994). Cooperative Systems Design, *Computing Journal*, 37(5), 357-366.

Sutcliffe, A.G., Maiden, N.A.M, Minocha, S. & Manuel, D. (1998). Supporting scenario-based requirements engineering, *IEEE Transactions on Software Engineering*, 24(12), 1072-1088.

Sutcliffe, A. (2003). *Scenario-based requirements engineering*, 11th IEEE International Requirements Engineering Conference, pp. 320-329, Monterey Bay, CA, USA.

Sutcliffe, A. (2002), *User-Centred Requirements Engineering: Theory and Practice*, Springer.

UrbanAPI project, (2011-2014), [Electronic version], Retrieved October 22, 2013, from http://www.urbanapi.eu/.

van Lamsweerde, A., Darimont, R. & Letier, E. (1998). Managing Conflicts in Goal-Driven Requirements Engineering, *IEEE Transactions on Software Engineering, Special Issue on Managing Inconsistency in Software Development*, 24(11), 908-926.

Wampler, B. & McNulty, S. (2011). Does Participatory Governance Matter? Exploring the Nature and Impact of Participatory Reforms, Comparative Urban Studies Project, Retrieved October 22, 2013, from http://www.wilsoncenter.org/publication/does-participatory-governance-matter.

Whittle, J. & Kruger, I.H. (2004). *A methodology for scenario-based requirements capture*, 26th International Conference on Software Engineering - W5S Workshop "Third International Workshop on Scenarios and State Machines: Models, Algorithms, and Tools", pp. 2-7. 25, Edinburgh, UK.

Winter, M., Brown, D. & Checkland, P. (1995). A role for soft systems methodology in information systems development, *European Journal of Information Systems*, 4(3), 130-142.

Wong, C. (2011), *Decision-making and problem-solving: Turning indicators into a double-loop evaluation framework*, in Hull, A., Alexander E.R., Khakee, A. & Woltjer, J. (Eds.), Evaluation for Participation and Sustainability in Planning, Routledge, London, UK.

Yigitcanlar, T., Velibeyoglu, K. & Baum, S. (2008). *Creative Urban Regions: Harnessing Urban Technologies to Support Knowledge City Initiatives*, ISBN 1599048388, IGI Global Snippet.