ePolicy

Engineering the Policy–making Life Cycle

Evaluation methodology

Document Type:	Deliverable
Dissemination Level:	PU (Public)
Editor:	Please specify the editor
Document state:	Final version (waiting for EC approval)
Document version:	
Contributing Partners:	all
Contributing WPs:	WP9
Estimated P/M (if applicable):	n.a.
Date of Completion:	27/02/2014
Date of Delivery to EC:	
Number of Pages:	35

ABSTRACT

This document is focused on defining the evaluation methodology for the ePolicy software. An analysis of the software component indicators reported in D2.3 is developed to give them a common structure in terms of target users, methods of testing, means of evaluation and the measurement of success. Moreover a contingency plan, specifying how to deal with the failure of a specific indicator or, in the worst case, of a whole component is defined. Additionally the document provides information on the possible testers of the software and the strategies foreseen to involve external evaluators, also including the questionnaires that will be used to collect their feedback.

SEVENTH FRAMEWORK PROGRAMME

The project is co-funded by the European Community under the Information and Communication Technologies (ICT) theme of the Seventh Framework Programme (FP7/2007-2013). Grant Agreement n°288147.

Authors of this document:

Daniele Sangiorgi[8], Lucia Mazzoni[8], Michela Milano[1], Federico Chesani[1], Stefano Bragaglia[1], Tina Balke[2], Nigel Gilbert[2], Peter G Johnson[2], Lars Kotthoff[3], Barry O'Sullivan[3], Luis Torgo[4], Tobias Ruppert[5], Tony Woods[7], Marco Gavanelli[9]

1: DISI, University of Bologna
2: The University of Surrey
3: 4C, University College Cork
4: INESC Porto
5: Fraunhofer IGD
6: Regione Emilia Romagna
7: PPA Energy
8: ASTER
9: ENDIF, University of Ferrara

Contents

This page has been intentionally left blank.

Executive Summary

This deliverable presents the evaluation methodology for the ePolicy software.

The evaluation is based on the indicators defined in D2.3 enriched with a set of common attributes. These attributes provide a description of each indicator and a specification of how each indicator "works", i.e. who will test it, how can it be measured, when it can be considered successful. Specifically for each indicator the following items are provided:

- The component to which the indicator refers
- Indicator type (functional/non functional)
- What are the methods of testing the indicator
- Which users will test the indicator
- Means of evaluation of the indicator
- A specification if the indicator is a "core indicator"

Particular effort have been devoted to the definition of a common set of values for the different attributes, in order to have a comparable set able to represent the different types of indicators for each of the software components.

A contingency plan is also included, which specifies how the project partners will deal with the failure of a specific indicator or, in the worst case, of a whole component.

The "core indicator" attribute identifies a sub set of functional indicators, representing those requirements needed for the correct operation of the component itself. Since the failure of even one core indicator of a certain component causes the failure of the whole component specific countermeasures have been reported.

The failure of a non-core indicator is considered individually and specific remedial actions have been defined.

Finally, the deliverable presents the users involvement strategy defining:

- A group of possible testers (Annex 1). This list is continuously updated and available at: https://docs.google.com/spreadsheet/ccc?key=0Ao3mrY7u3thmdGdBWUpSbnVzQkNGZkZLSjV4cDNuTmc&usp=sharing
- A strategy to involve external evaluators (Section 4). Specific actions are foreseen at the next Samos summit on ICT-enabled Governance and the associated 2nd International Summer School on Open and Collaborative Governance (July 2014) and at the ePolicy event that will be organized in Bologna (Autumn 2014).
- The tools to support the testing phase , i.e. the user guidelines and the questionnaires used (Section 5 and Annex 2)

1 Introduction

This document is focused on defining the evaluation methodology for the ePolicy software. It specifies how the evaluation will be conducted and the related contingency plan.

Section 2 reports on the analysis of the indicators presented in D2.3, defining a common structure for these indicators, specifying the users, the methods of testing, the means of evaluation and the related measure of success.

Section 3 includes the contingency plan, which specifies how project partners will deal with the failure of a whole component or of a specific indicator.

Section 4 describes the actions and strategies to be adopted to involve external evaluators.

Section 5 outlines references to documents which will provide user guidelines. Full user guidelines will be available at the end of the software development phase planned for June, 2014, as the GANTT chart of the project indicates.

Annex 1 provides a list of possible testers at a particular date i.e. February 26th 2014. This list is available on line and continuously updated.

Annex 2 shows the list of questions to which testers will be asked to respond during and after testing sessions.

2 Indicators Evaluation

The indicators reported in D2.3 have been analyzed with the objective of identifying a set of common attributes which describe each indicator, including specifying how each indicator "works", i.e. who will test it, how it can be measured, and when it can be considered successful.

The list of attributes identified for each indicator is the following one:-

- The component to which the indicator refers
- Indicator type (functional/non functional)
- The methods of testing the indicator
- The users that will test the indicator
- The means of evaluation of the indicator
- The specification if the indicator is a "core indicator"

For each of these attributes a set of possible values has been defined. These are in line with the content of D2.3 so that there is a common structure for the indicators referring to the different components of the ePolicy software.

Specifically the identified attributes, and the related values, are:

- **Component**[1]**.** This specifies the component to which a particular indicator refers. The full list of the software components are as follows:
  - Global Optimizer
  - Incentive design module
  - Social Simulator
  - Opinion mining
  - Visualization Modules
  - Software architecture
- I**ndicator Type**. This specifies if the indicator is a functional or non-functional one.
  - Functional: an indicator which represents the operations and activities that a component must be able to perform. i.e. the core functionalities
  - Non Functional: an indicator which represents how a component is able to do the operations and activities that performs. i.e. performance, scalability, user friendliness
- **Test procedure**:
  - questionnaire: a set of questions submitted to the testers of the ePolicy software. The questions are provided in the "ANNEX 2 – Questionnaires for testing"
  - analysis/test of the code: analysis of the source code of the software and/or run of tests in order to verify if the functional/non-functional specifications are achieved.
  - analysis of output parameters: analysis of the output to verify if the results obtained are compliant with what is expected.
  - Laboratory Experiment: users have to solve predefined tasks with the application. The success rate and time needed to complete the task will be measured. In many cases an additional questionnaire will be handed out so that users can report their subjective views on the usability and friendliness of the software.
- **Tester groups**:
  - Developer: ePolicy software developer.
  - Policy Maker[2]: the main user of the ePolicy system, who may use the system and run the software, specifying different input parameters
  - Environmental Expert: Domain expert and expert user of the system. As well as the rights of the policy maker user, this user may also specify certain configuration parameters which are specific to the domain under investigation (e.g. the values of coaxial matrices).
  - Public user: other users, non IT experts. These parties will contribute to the testing of the software (in particular the visual interfaces).
- **Means of Evaluation**:
  - Straightforward: it is immediate to verify if the indicator is fulfilled or not

[1] See full description of the components in D2.3

[2] The full specifications of what Policy Makers and Environmental Experts are allowed to do with the software application can be found in D3.3

    - Satisfaction of 60% of users: the indicator is considered fulfilled if at least the 60% of the testers have given a positive feedback
    - Other means of evaluation: specific for each indicator. Specified in the tables below.
- **Core indicator**: core indicators are a sub-set of functional indicators; they represent those requirements needed for the full operation of the component itself.

In the following tables the indicators list is reported for each component with the related attributes.

2.1 Global Optimizer

Indicator	Indicator Type	Test procedure	Testers group	Means of Evaluation	Core indicator?
List of functionality required by the environmental expert: - Set regional max energy per source - Change environmental matrices - Change primary/secondary activity matrix - Define constant per activity - Min and max energy - State objective functions - Set total energy - Run system - Obtain results	Functional	Analysis/test of the code/software	Developers	Number of functional requirements fulfilled. If: N<60%: unsatisfactory; 60%<N<80%: sufficient; N>80%: satisfactory	Yes
List of functionality required by the policy maker: - Min and max energy - State objective functions - Set total energy - Run system - Obtain results	Functional	Analysis/test of the code/software	Developers	Number of functional requirements fulfilled. If: N<60%: unsatisfactory; 60%<N<80%: sufficient; N>80%: satisfactory	Yes
Scalability, performance and efficiency	Non Functional	Analysis/test of the code/software	Developers	Response time less than five minutes	No
Development costs and time	Non Functional	Analysis of output parameters	Developers	Variation of time and cost is lower than the 5% with respect to the budgeted time and cost.	No
Accuracy, precision and re-configurability	Non Functional	Questionnaire	Policy Maker Env. Expert	Success: more than the 60% of users are satisfied	No
Ease of use	Non Functional	Questionnaire	Policy Maker Env. Expert	Success if more than the 60% of users consider the component easy to use	No
Problems occurred while using the component	Non Functional	Questionnaire	Policy Maker Env. Expert	Success if more than the 60% of the problems identified can be rapidly solved	No

Adequacy of the service (Rate 1 to 10)	Non Functional	Questionnaire	Policy Maker Env. Expert	Success if the mean of the ratings is greater than 6 (on a scale from 1 to 10)	No

2.2 Social simulator

Indicator	**Indicator Type**	**Test procedure**	**Testers group**	**Means of Evaluation**	**Core indicator?**
Realistically presents likely adoption of photovoltaic panels under different policy instrument settings chosen	Functional	Analysis of output parameters	Developers	Comparison with past data and other beliefs about adoption patterns	Yes
The users need to be able to select and specify which of the above mentioned policy instruments they wish to model	Functional	Analysis/test of the code/software	Developers	Straightforward	No
The users need to be able to choose whether they wish to view the whole Emilia-Romagna region or only sub-regions	Functional	Analysis/test of the code/software	Developers	Straightforward	No
The social simulator must provide information that allows its users to evaluate the different policy instrument implementation strategies	Non Functional	Analysis of output parameters	Developers	Meeting previous input of users (from questionnaire outlined in D4.1))	No
The social simulator should allow the use of GIS data for the setup of decision making entities and the environment they act in	Non Functional	Analysis/test of the code/software	Developers	Straightforward	No
The social simulator setup is grounded on empirical data collected by interviews and an online questionnaire	Non Functional	Analysis/test of the code/software	Developers	Straightforward	No
The social simulator component must be developed using open source tools	Non Functional	Analysis/test of the code/software	Developers	Straightforward	No
The social simulator software should be platform independent	Non Functional	Analysis/test of the code/software	Developers	Straightforward	No
Ease of use	Non Functional	Questionnaire	Policy Maker Env. Expert	Success if more than the 60% of users consider the component easy to use	No

Problems occurred while using the component	Non Functional	Questionnaire	Policy Maker Env. Expert	Success if more than the 60% of the problems identified can be rapidly solved	No
Adequacy of the service (Rate 1 to 10)	Non Functional	Questionnaire	Policy Maker Env. Expert	Success if the mean of the ratings is greater than 6 (on a scale from 1 to 10)	No

2.3 Incentive design

Indicator	**Indicator Type**	**Test procedure**	**Testers group**	**Means of Evaluation**	**Core indicator?**
Incentive design component must output incentive designs	Functional	Analysis of output parameters	Policy Maker	Straightforward	Yes
Incentive design component must optimise incentive designs	Functional	Analysis of output parameters	Policy Maker	Straightforward	No
Incentive design component must consider goals for energy production and budget constraints.	Functional	Analysis/test of the code/software	Policy Maker	The incentive design will specify figures for those constraints that will allow a judgment to be made as to whether they have been fully considered	Yes
Incentive design component must allow users to explore different allocation scenarios	Functional	Analysis/test of the code/software	Policy Maker	Straightforward	No
Incentive design component must allow the consideration of policy decisions (through specific input boxes/elements in the user interface that allows the input of items such as a budget restriction)	Functional	Analysis of output parameters	Policy Maker	Straightforward	No
Incentive design component must interface with the social simulator	Functional	Analysis/test of the code/software	Developers	Straightforward	No
Incentive design component must interface with the ePolicy system	Functional	Analysis/test of the code/software	Developers	Straightforward	No
Incentive design component must be easy to use	Non Functional	questionnaire	Policy Maker	Success if more than the 60% of users consider the component easy to use	No

Incentive design component must provide solutions within a reasonable time.	Non Functional	Analysis/test of the code/software	Policy Maker	Test of the component with a number of different scenarios. Success if response time less than 2 minutes for each scenario	No
Incentive design component must provide information that allows the user to evaluate incentive designs	Non Functional	Questionnaire	Policy Maker	Success if more than the 60% of users are satisfied	No
Incentive design component must be installable with reasonable effort	Non Functional	Questionnaire	Policy Maker	Success if more than the 60% of users are satisfied	No
Incentive design component must not depend on third-party components that incur additional costs	Non Functional	Analysis/test of the code/software	Developers	Straightforward	No
Problems occurred while using the component	Non Functional	Questionnaire	Policy Maker	Success if more than the 60% of the problems identified can be rapidly solved	No
Adequacy of the service (Rate 1 to 10)	Non Functional	Questionnaire	Policy Maker	Success if the mean of the ratings is greater than 6 (on a scale from 1 to 10)	No

2.4 Opinion mining

Indicator	**Indicator Type**	**Test procedure**	**Testers group**	**Means of Evaluation**	**Core indicator?**
The opinion mining component has to give a set of sentiment scores for a pre-defined set of topics.	Functional	Analysis of output parameters	Developers	Straightforward	Yes
The opinion mining component should be able to classify new documents regarding the expressed sentiment on a set of pre-defined topic	Functional	Analysis of output parameters	Developers	Straightforward	Yes

The opinion mining component must be able to aggregate the classifications of new text documents into an overall sentiment concerning the topics, on a certain time scale (e.g. weekly, daily, etc.).	Functional	Analysis of output parameters	Developers	Straightforward	Yes
The opinion mining component should be able to accept as input a set of topics of interest for sentiment analysis,	Functional	Analysis of output parameters	Developers	Straightforward	Yes
Development of a set of models that are able to classify documents concerning opinions on a set of predefined topics	Functional	Analysis of output parameters	Developers	Standard evaluation methodologies for predictive models to obtain statistically significant measures of the accuracy of these models in terms of their classification tasks.	Yes
Ability to fetch from the given web sources, documents that are relevant for the selected topics of interest	Functional	Analysis of output parameters	Developers	Straightforward	Yes
Software usability, in terms of being able to present the user with the correct and useful information that they are looking for	Functional	Questionnaire	Policy Maker Env. Expert	Success if more than the 60% of users are satisfied	No
The opinion mining component should be implemented using only free software	Non Functional	Analysis/test of the code/software	Developers	Straightforward	No
The opinion mining component should be easy to adapt to new domains	Non Functional	Questionnaire	Developers	Success if more than the 60% of users are satisfied	No
The opinion mining component should facilitate extensions to the set of topics and/or web sites.	Non Functional	Questionnaire	Developers	Success if more than the 60% of users are satisfied	No
The opinion mining component should provide easy ways of exploring the tendency of the sentiment scores	Non Functional	Questionnaire	Policy Maker Env. Expert	Success if more than the 60% of users are satisfied	No
The opinion mining component should provide means to drill down the scores to the actual messages that lead to the scores	Non Functional	Questionnaire	Policy Maker Env. Expert	Success if more than the 60% of users are satisfied	No
Ease of use	Non Functional	Questionnaire	Policy Maker Env. Expert	Success if more than the 60% of users consider the component easy to use	No

Problems occurred while using the component	Non Functional	Questionnaire	Policy Maker Env. Expert	Success if more than the 60% of the problems identified can be rapidly solved	No
Adequacy of the service (Rate 1 to 10)	Non Functional	Questionnaire	Policy Maker Env. Expert	Success if the mean of the ratings is greater than 6 (on a scale from 1 to 10)	No

2.5 Visualization

Indicator	**Indicator Type**	**Test procedure**	**Testers group**	**Means of Evaluation**	**Core indicator?**
The visual interfaces have to provide access to the components (global optimizer, social simulator, opinion mining)	Functional	Laboratory Experiment	Policy Maker Env. Expert Public user	Measure success rate and time needed to solve predefined tasks derived from requirement analysis. Success if more than 60% of tasks successfully completed	Yes
The users should be enabled to visual-interactively define input parameters for each component	Functional	Laboratory Experiment	Policy Maker Env. Expert Public user	Measure success rate and time needed to solve predefined tasks derived from requirement analysis. Success if more than 60% of tasks successfully completed	Yes
The users should be enabled to view the output data of the technical components and analyze this data	Functional	Laboratory Experiment	Policy Maker Env. Expert Public user	Measure success rate and time needed to solve predefined tasks derived from requirement analysis. Success if more than 60% of tasks successfully completed	Yes

Visual encodings	Non Functional	Questionnaire	Policy Maker Env. Expert Public user	In these questionnaires the users are asked whether they understand the visual designs and whether they intuitively know how to use the interfaces. Success if more than an average rate of over 60% approves the usability of the visualization component	No
Interaction designs	Non Functional	Questionnaire	Policy Maker Env. Expert Public user	In these questionnaires the users are asked whether they understand the visual designs and whether they intuitively know how to use the interfaces. Success if more than an average rate of over 60% approves the usability of the visualization component	No
Ease of use	Non Functional	Questionnaire	Policy Maker Env. Expert Public user	Success if more than the 60% of users consider the component easy to use	No
Problems occurred while using the component	Non Functional	Questionnaire	Policy Maker Env. Expert Public user	Success if more than the 60% of the problems identified can be rapidly solved	No
Adequacy of the service (Rate 1 to 10)	Non Functional	Questionnaire	Policy Maker Env. Expert Public user	Success if the mean of the ratings is greater than 6 (on a scale from 1 to 10)	No

2.6 Software architecture

Indicator	Indicator Type	Test procedure	Testers group	Means of Evaluation	Core indicator?
User authentication: secure mechanisms to identify user interactions, based on the username/password model	Functional	Analysis/test of the code/software	Policy Maker Env. Expert	Success if at least four out of the five requirements are implemented and made available to the single components.	No
Support an Access Control List method for users/roles and components/functionalities	Functional	Analysis/test of the code/software	Policy Maker Env. Expert		No
Provide a web-based access to the components	Functional	Analysis/test of the code/software	Policy Maker Env. Expert		No
Support users' "work sessions" distributed over time and geographical locations	Functional	Analysis/test of the code/software	Policy Maker Env. Expert		No
Provide data persistence facilities: users and components might need to save partial/complete computation results and plans	Functional	Analysis/test of the code/software	Policy Maker Env. Expert		No
Scalability, performance and efficiency:	Non Functional	Analysis/test of the code/software	Developers	Architecture added time overhead less than 10% of the mean time for using each single component.	No
Accuracy, precision and re-configurability	Non Functional	Questionnaire	Policy Maker Env. Expert	Success if more than the 60% of users are satisfied	No
Cost of development and time of development	Non Functional	Analysis of output parameters	Developers	Success if the costs used for development do not exceed what is foreseen in the project by more than 15%	No

3 Contingency Plan

The contingency plan defines remedial actions should some of the indicators fail during the tests. In the following sections these will be specified for each component as follows:

- Remedial action/actions if the non-core indicators are not fulfilled
- What to do if the whole component fails (i.e. at least one core indicators fails)

3.1 Global Optimizer

3.1.1 Remedial action if the non-core indicators are not fulfilled

It should be noted that users have been consulted before developing indicators/requirements (presented in D2.3) so the user input was built into the development process. As such many of them are simply met in the basic design approach.

Indicator (non-core)	Remedial action if failed
Scalability, performance and efficiency	These indicators have already been tested in deliverable 3.3, so they cannot fail.
Development costs and time	
Accuracy, precision and re-configurability	If one of these fails to achieve a good score, the component will need to be redesigned based on the suggestions within the feedback. However, these indicators have been tested with the first prototype. It has since been improved and new functionalities identified after the first round of testing have been added. These indicators will be considered in the next test that will be contained in D3.5
Ease of use	
Problems occurred while using the component	
Adequacy of the service (Rate 1 to 10)	

3.1.2 What to do if the whole component is failed (i.e. core indicators failed)

List of core indicators for the component:

- List of functionality required by the environmental expert (Set regional max energy per source; Change environmental matrices; Change primary/secondary activity matrix; Define constant per activity; Min and max energy; State objective functions; Set total energy; Run system; Obtain results)
- List of functionality required by the policy maker (Min and max energy; State objective functions; Set total energy; Run system; Obtain results)

Technical failure during use:

If there is a data base, the closest plan could be found, but this requires further research.

User-driven failure

If the users have any problems with the tool during testing and would like to explore changes they will be put in touch with the developers to identify the source of the problems.

3.2 Social simulator

3.2.1 Remedial action if the non-core indicators are not fulfilled

Also for this component users have been consulted before developing indicators/requirements (presented in D2.3) so this user input was built into the development process. As such many of them are simply met in the basic design approach. The questionnaire given to users for this purpose is in the appendix of D4.1.

Indicator (non-core)	Remedial action if failed
The users need to be able to select and specify which of the above mentioned policy instruments they wish to model	Built into the basic design of the tool already, so cannot fail.
The users need to be able to choose whether they wish to view the whole Emilia-Romagna region or only sub-regions	Built into the basic design of the tool already, so cannot fail.
The social simulator must provide information that allows its users to evaluate the different policy instrument implementation strategies	Information requested by users in the initial questionnaire already included. If there is any additional information that users would like that is not included, then they will be put into contact with the developers to explore possible actions.
The social simulator should allow the use of GIS data for the setup of decision making entities and the environment they act in	Built into the basic design of the tool already, so cannot fail.
The social simulator setup is grounded on empirical data collected by interviews and an online questionnaire	Interviews and survey carried out and built into the basic design of the tool already, so cannot fail.
The social simulator component must be developed using open source tools	Has been done, so cannot fail.
The social simulator software should be platform independent	Has been done, so cannot fail.
Ease of use	Small fixes can be made easily/quickly by updating code/interface. Larger changes may require longer re-development.
Problems occurred while using the component	Small fixes can be made easily/quickly by updating code/interface. Larger changes may require longer re-development.
Adequacy of the service (Rate 1 to 10)	Small fixes can be made easily/quickly by updating code/interface. Larger changes may require longer re-development.

3.2.2 What to do if the whole component is failed (i.e. core indicators failed)

List of core indicators for the component:

- Realistically presents likely adoption of photovoltaic panels under different policy instrument settings chosen

Technical failure during use

In the final web service the results of the simulator will not be run live, but stored in a database. Hence failure of the individual component is not possible if the overall system is working. The

database is in effect already a backup for the tool not working. If users are using the stand-alone tool and they have technical issues they should contact the developers for support.

User-driven failure

If the users have any problems with the tool during testing and would like to explore changes they will be put in touch with the developers to identify the source of problems.

3.3 Incentive design

3.3.1 Remedial action if the non-core indicators are not fulfilled

Indicator (non-core)	Remedial action if failed
Incentive design component must optimise incentive designs	These indicators have already been achieved in previous deliverables and will be retested to verify their behaviour in the overall system
Incentive design component must allow users to explore different allocation scenarios	
Incentive design component must allow the consideration of policy decisions (through specific input boxes/elements in the user interface that allows the input of items such as a budget restriction)	
Incentive design component must interface with the social simulator	
Incentive design component must interface with the ePolicy system	Must implement interface to ePolicy system.
Incentive design component must be easy to use	Take user feedback into account and redesign component user interface.
Incentive design component must provide solutions within a reasonable time.	Increase hardware requirements, tune performance and pre-compute/cache results.
Incentive design component must provide information that allows the user to evaluate incentive designs	This indicator has been achieved already in previous deliverables
Incentive design component must be installable with reasonable effort	Take user feedback into account and redesign installation process.
Incentive design component must not depend on third-party components that incur additional costs	This indicator has been achieved already in previous deliverables and will be retested.
Problems occurred while using the component	Small fixes can be made easily/quickly by updating code/interface. Larger changes may require longer re-development.
Adequacy of the service (Rate 1 to 10)	Small fixes can be made easily/quickly by updating code/interface. Larger changes may require longer re-development.

3.3.2 What to do if the whole component is failed (ie. core indicators failed)

List of core indicators for the component:

- Incentive design component must output incentive designs
- Incentive design component must consider goals for energy production and budget constraints.

Deliverables 5.2 and 5.3 already implement working prototypes so the former indicators are fulfilled and the component is proved to work.

Technical failure during use:
Fall back to pre-computed results stored in a database.

User-driven failure
If the users have any problems with the tool during testing and would like to explore changes they will be put in touch with the developers to identify the source of problems.

3.4 Opinion mining

3.4.1 Remedial action if the non-core indicators are not fulfilled

Indicator (non-core)	Remedial action if failed
Software usability, in terms of being able to present the user with the correct and useful information that they are looking for	Use the feedback from questionnaires to improve the software
The opinion mining component should be implemented using only free software	Already done, so it cannot fail
The opinion mining component should be easy to adapt to new domains	Use the feedback from questionnaires to improve the way the software handles domain settings
The opinion mining component should facilitate extensions to the set of topics and/or web sites.	Use the feedback from questionnaires to improve the way the software handles domain settings
The opinion mining component should provide easy ways of exploring the tendency of the sentiment scores	Already done, so it cannot fail
The opinion mining component should provide means to drill down the scores to the actual messages that lead to the scores	Already done, so it cannot fail
Ease of use	Use the feedback from questionnaires to improve the software
Problems occurred while using the component	Use the feedback from questionnaires to improve the software
Adequacy of the service (Rate 1 to 10)	Use the feedback from questionnaires to improve the software

3.4.2 What to do if the whole component is failed (ie. core indicators failed)
List of core indicators for the component:

- The opinion mining component has to give a set of sentiment scores for a pre-defined set of topics.
- The opinion mining component should be able to classify new documents regarding the expressed sentiment on a set of pre-defined topic
- The opinion mining component must be able to aggregate the classifications of new text documents into an overall sentiment concerning the topics, on a certain time scale (e.g. weekly, daily, etc.).
- The opinion mining component should be able to accept as input a set of topics of interest for sentiment analysis,

- Development of a set of models that are able to classify documents concerning opinions on a set of predefined topics
- Ability to fetch from the given web sources, documents that are relevant for the selected topics of interest

Technical failure during use:

Policy makers (standard users of the opinion mining (OM) module) will interact with the OM prototype through a visual interface that uses information stored in a database. This information (the opinion scores derived from the opinion mining algorithms) was previously obtained, so the only possible failure in this standard usage of the OM module is that the overall web interface of the decision support system breaks down, or the database management system somehow fails, which is a highly unlikely to happen, but if it did occur would require a system administrator to take corrective actions.

User-driven failure:

Policy makers will not directly interact with the opinion mining (OM) software prototype. Their interaction will be with the visual interface of the overall ePolicy decision support system that fetches the necessary information from a data base to provide visual feedback concerning the trends of the public opinion concerning the selected topics. In this context, it is not possible for a standard user to drive the OM module to failure.

3.5 Visualization

3.5.1 Remedial action if the non-core indicators are not fulfilled

Indicator (non-core)	**Remedial action if failed**
Visual encodings	• Include suggestions by users in design. • Parts of the visualization tool have already been tested after the first round of implementation and been approved
Interaction designs	
Ease of use	
Problems occurred while using the component	
Adequacy of the service (Rate 1 to 10)	

3.5.2 What to do if the whole component is failed (ie. core indicators failed)

List of core indicators for the component:

- The visual interfaces have to provide access to the components (global optimizer, social simulator, opinion mining)
- The users should be enabled to visual-interactively define input parameters for each component
- The users should be enabled to view the output data of the technical components and analyze this data

Technical failure during use:

Technical failure of the component may occur due to failures of the connected technical components (see previous sections 3.1to 3.4), or of the visualization component itself. In the first

case, the user is referred to the other components (e.g. databases could be used as a data sink). In the latter case, the developers will have to be contacted. Moreover, alternative visual interfaces to the technical components may be designed by third parties or the technical components could be used without a visual interface.

User-driven failure:
If the users have any problems with the tool and would like to explore changes they will need to be put into contact with the developers.

3.6 Software architecture

3.6.1 Remedial action if the non-core indicators are not fulfilled

Indicator (non-core)	**Remedial action if failed**
User authentication: secure mechanisms to identify user interactions, based on the username/password model	The current architecture envisages the management of security aspects such as authentication and the access control list as part of the architecture itself. If such an indicator were not to be achieved, a simpler security model would be implemented. In particular, since the application is web-based, two different alternatives can be explored: security based on the web app container, and security based on a restricted access network. Both the alternatives can exploit the existing technology.
Support an Access Control List method for users/roles and components/functionalities	
Provide a web-based access to the components	Web-based access has already been provided for some components. Hence the indicator is partly fulfilled. However all the components can be used as standalone software running locally on workstation-size machines, and can provide the output in terms of CSV-style data. Such data can be then easily analyzed locally.
Support users' "work sessions" distributed over time and geographical locations	The current architecture will consider also local-only, standalone interactions, since all the components as well as the architecture can be executed locally.
Provide data persistence facilities: users and components might need to save partial/complete computation results and plans	The current architecture envisages the storage of user-related information and data within a database. If the indicator were to be failed, a remedial solution would consist of partly redesigning the architecture so as to let the user to directly store the data locally on their computer, in the form of human readable files.
Scalability, performance and efficiency:	The current architecture is based on a distributed environment where different hosts provide the component services. If scalability and performances become problematic, the introduction of different physical hosts and clustering techniques will be adopted to mitigate the problem.

Accuracy, precision and re-configurability	Small fixes can be made easily/quickly by updating the architecture when needed. Larger changes may require longer re-development.
Cost of development and time of development	At the current project stage the expenses are in line with the budget provisioned for the architecture. If needed, further personnel (developers) can be added to the work package development team.

3.6.2 What to do if the whole component is failed (ie. core indicators failed)

Software architecture has no core indicators and users are not foreseen to directly use this component

3.7 Overall system

The overall system is the integration of the previously described components, through the architecture. The remedial actions to be undertaken should a component fail or if the integrated architecture fails are described in the previous sections and apply for the overall system.

4 Involvement of External Evaluators

Many participants in the project (including those from RER, PPA, ASTER and other stakeholders like ARPA) are being used to evaluate the components that make it up using a series of functional and non-functional indicators, as described in section 2.

It is vital that this evaluation includes assessments of the "user friendliness" of these components. Hence for each of them the following indicators have been included:-

- Is the component easy to use?
- What problems occurred whilst using the component?
- What is the overall rating (from one to ten) of the adequacy of the component in meeting the needs of the user?

However it is also important to assess the reactions of other stakeholders - who have had little or no prior contact with the project - to the components. This again particularly relates to the practicality and ease of use of the delivered software. It is intended that these stakeholders will be policy and environmental experts who would be most likely to consider using such an approach for handling complex policy issues.

Similar indicators to those described above can be used to do this and the method of assessment would also again be by means of a questionnaire. This would be similar to those questionnaires outlined in Annex 2 although in some cases it may be appropriate to shorten and simplify them as the contact time with the individuals concerned is likely to be much less than is the case with those from organizations that are participating in the project.

There are serious challenges in achieving a sufficient level of involvement from potential users of the ePolicy software and methodology especially from "external" parties who have not previously been involved with or even aware of the project. Thus a creative approach is required so that such involvement is maximized. This will be achieved in a number of ways. Firstly it is intended to disseminate information about ePolicy by means of the organization of or participation in a number of events during the closing stages of the project, and as the software that makes up the components matures. For example a number of ePolicy participants are intending to attend the 5th Samos Summit on ICT-enabled Governance and the associated 2nd International Summer School on Open and Collaborative Governance in Greece in July 2014. This will provide the opportunity to both demonstrate the ePolicy components and to get immediate, structured, external stakeholder feedback. Similarly a further ePolicy organized stakeholder event is being planned for autumn 2014 in Bologna, Italy where again such feedback will be sought. Other opportunities will be exploited as they are identified.

A second approach is to increase the level of involvement of the various contacts that individual ePolicy participants have had with parties that have expressed interest in the project so far. In the absence of stable and robust software developing a continued interest in the project is inevitably problematic. With the completion of the software modules it should be possible to stimulate further interest from such parties. This will require individual contact with them almost always at their own workplaces as this will increase the chance of participation. Again a questionnaire approach will be used – simplified as much as possible so as to again, as far as possible, ensure involvement.

The methodology that has been conceived to enable potential stakeholders, not closely involved in the ePolicy project, to test the system is to create a short, fairly simple and light-touch presentation of the project, containing the following sections:-

- Overall ePolicy concept and motivation
- Description of the main components
- Brief demonstration of the software prototype

The final demonstration explains how to use the system which is being tested (either the whole architecture or a single component) and enables the user to understand the main steps that should be carried out to use it. After this presentation which is likely to take approximately 20-30 minutes, users will be allowed to use the software on their own to gain familiarity with it and explore its potential.

At the end of this period of familiarization – which again is expected to last around 20-30 minutes – participants will be asked to complete a short questionnaire in regard to the software that they have been using including especially its ease of use.

The resulting completed questionnaires will be summarized and the results included in the corresponding deliverable where the evaluation of the related component is contained

5 Guidelines for users

Specific guidelines for each component will be developed when each is completed, in the relevant component deliverable. At present a set of basic instructions is available (for the first version prototypes) and these can be found in the following documents:

- GLOBAL OPTIMIZER: basic instructions have been reported in D3.2.
- SOCIAL SIMULATOR and INCENTIVE DESIGN: basic instructions for both components have been reported in Deliverable D5.3
- OPINION MINING: As the interaction of policy makers with the opinion mining results will be made through the visualization component, specific instructions on this interaction will be provided by WP7 deliverables. Specific instructions for developers will be reported in D6.4
- VISUALIZATION: basic instructions have been reported in D7.1 and D7.3.

6 Conclusions

This deliverable has presented the evaluation methodology for the ePolicy software.

The evaluation is based on the indicators presented in D2.3 and enriched with the attributes presented in section 2, which specify the target users, the methods of testing, the means of evaluation and the measurement of success.

Particular effort have been devoted to the definition of a common set of values for the different attributes, in order to have a limited and comparable set able to represent the different types of indicators for each of the software components.

The contingency plan specifies how project partners will deal with the failure of a specific indicator or, in the worst case, of a whole component. A sub-set of indicators, called Core Indicators, have been defined, representing those requirement needed for the full operation of the component itself. A failure of a core indicator implies the failure of the whole component, and the cases "Technical failure during use" and "User-driven failure" have taken this into consideration. The failure of a non-core indicator is considered individually and specific remedial actions have been defined.

Finally the deliverable presents the user involvement strategy, defining groups of possible testers (Annex 1); a strategy to involve external evaluators (Section 4) and the tools to support the testing phase, i.e. the user guidelines and the questionnaire used (Sections 5 and Annex 2).

7 ANNEX 1 – List of testers

The List of possible testers is continuously updated and reported in:

https://docs.google.com/spreadsheet/ccc?key=0Ao3mrY7u3thmdGdBWUpSbnVzQkNGZkZLSjV4cDNuTmc&usp=sharing

Here follows and extraction of these contacts on February 26th

N.	Institution/Enterprise/Public Body	Country	Contact Name	Telephone	Mobile	Email
1	Adria Energy ESCO	Italy	Ing. Roberto Renzi	0541 833160 0541 833166		robertorenzi@umpi.it a.capponi@adriaenergy.it
2	Energate	Italy	Freschi Marco	0522 849616	349 8437094	freschimarco@energate.it
3	Ecosyntesi	Italy	Luigi Stangarone	051 6466249		info@ecositnthesy.it marketing@ecosynthesi.it
4	Nuova energia	Italy	Andrea Bonzi	051 0827204		tecnico@nuova-energia.it
5	Bioaus	Italy	Pietro Fiumana	0543 34171	335 7845500	pf@bioaus.eu
6	Impronte Soc. Coop.	Italy	Luca Orioli		328 4149764	luca.orioli@cooperativaimpronte.it
7	Solare sociale	Italy	Lorenzo Cardinali	051 803668	348 5319940	
8	Termal srl	Italy	Luisa Ciordinik	051 4133111		info@termal.it, luisa.ciordinik@termal.it
9	Techno GF srl	Italy	Marcello Degli Esposti	051 847552		marcello.degliesposti@technogf.com
10	Siet srl	Italy	Signor Brugnatti	0532 744821		info@sietonline.it
11	Marano Solar	Italy		0541 755151		fabio.peruzzi@maranosolar.it
12	Beghelli Servizi snc	Italy	Parazza Simona	051 9660411		marketing@beghelli.it
13	Y.U.PPIES' SERVICES Srl	Italy	Ing Elena Braglia	059 282727		e.braglia@yuppies.it

14	Bertolini prodi & c	Italy	Marco Prodi	051 6951566		m.prodi@officioarchitettura.it
15	AESS - Agenzia per Energia e Sviluppo sostenibile	Italy	Claudia Carani Lisa Sentimenti	059 452511		ccarani@aess-modena.it lsentimenti@aess-modena.it
16	Centro CISA	Italy	Sergio Palmieri Marco Odaldi	0534 24022	348 4108641 329 4122589	palmieri@centrocisa.it info@centrocisa.it
17	Ervet	Italy	Fabrizio Tollari	051 6450474	349 0626039	ftollari@ervet.it
18	ARPA - SERVIZIO IDRO-CLLIMA	Italy	Vittorio Marletto	051 6497564	335 7956636	vmarletto@arpa.emr.it
19	ENEL Ingegneria e Ricerca	Italy	Sandra Scalari			sandra.scalari@enel.com
20	Comune di Genova	Italy	Alessandra Risso			arisso@comune.genova.it
21	Minister of Economic Development	Italy	Oriana Coccu			oriana.cuccu@dps.gov.it
22	RSE	Italy	Cristina Cavicchioli			cristina.cavicchioli@rse-web.it
23	ENEA	Italy	Roberto Del Ciello			roberto.delciello@enea.it
24	Emilia-Romagna Region	Italy	Attilio Raimondi Anna Maria Linsalata			araimondi@regione.emilia-romagna.it alinsalata@regione.emilia-romagna.it"
25	ASTER	Italy	Stefano Valentini			stefano.valentini@aster.it
26	PPA ENERGY	UK	Tony Woods			tony.woods@ppaenergy.co.uk
27	DECC	UK	Craig Lucas			craig.lucas@decc.gsi.gov.uk
28	DECC	UK	Dr Sarah Ulliott	0300 068 6658		Sarah.Ulliott@decc.gsi.gov.uk
29	DEFRA	UK	Davinder Lall			davinder.lail@defra.gsi.gov.uk
30	DEFRA	UK	Paul Nunn			Paul.Nunn@defra.gsi.gov.uk

8 ANNEX 2 – Questionnaires for testing

In this Annex are reported all the questionnaires foreseen to test the ePolicy indicators. These questionnaires could be updated as the project partners get experience with their use and gain further experience with the software as it matures.

8.1 Global Optimizer

Indicator	**Testers group**	**Questionnaire**
Accuracy and precision	Env. Expert	Please indicate which statement represents how you feel about the tool after you have used it. i. It is accurate in the sense that it provides precise information about the plans and the impact of the plans proposed 1. Strongly agree 2. Agree 3. Neutral 4. Disagree 5. Strongly disagree 6. Not sure in case you are not satisfied with the precision or accuracy, can you suggest how to improve it?
Reconfigurability	Env. Expert	Please indicate which statement represents how you feel about the tool after you have used it. i. It is easily configurable? 1. Strongly agree 2. Agree 3. Neutral 4. Disagree 5. Strongly disagree 6. Not sure in case you are not satisfied with the reconfigurability , can you suggest how to improve it?

Ease of use	Policy Maker+Env. Expert	Please indicate which statement represents how you feel about the tool after you have used it. i. It is simple to use 1. Strongly agree 2. Agree 3. Neutral 4. Disagree 5. Strongly disagree 6. Not sure ii. It is user friendly 1. Strongly agree 2. Agree 3. Neutral 4. Disagree 5. Strongly disagree 6. Not sure In case you are not satisfied with the easiness of use, can you suggest how to improve it?
Problems occurred while using the component	Policy Maker+Env. Expert	Please describe any problems that occurred during your use of the tool (either technical or related to your understanding of its operation). i. Open text box
Adequacy of the service (Rate 1 to 10)	Policy Maker+Env. Expert	Please rate the adequacy of the tool in your opinion (1 = Meets no requirements, and 10 = Meets all requirements) i. 1-10 scale ii. Comment box

8.2 Social simulator

Indicator	**Testers group**	**Questionnaire**
Ease of use	Policy Maker+Env. Expert	Please indicate which statement represents how you feel about the tool after you have used it. i. It is simple to use 1. Strongly agree 2. Agree 3. Neutral 4. Disagree 5. Strongly disagree 6. Not sure ii. It is user friendly 1. Strongly agree 2. Agree 3. Neutral 4. Disagree 5. Strongly disagree 6. Not sure In case you are not satisfied with the easiness of use, can you suggest how to improve it?

Problems occurred while using the component	Policy Maker+Env. Expert	Please describe any problems that occurred during your use of the tool (either technical or related to your understanding of its operation). i. Open text box
Adequacy of the service (Rate 1 to 10)	Policy Maker+Env. Expert	Please rate the adequacy of the tool in your opinion (1 = Meets no requirements, and 10 = Meets all requirements) i. 1-10 scale ii. Comment box

8.3 Incentive design

Indicator	**Testers group**	**Questionnaire**
Incentive design component must be easy to use	Policy Maker+Env. Expert	Please indicate which statement represents how you feel about the tool after you have used it. i. It is simple to use 1. Strongly agree 2. Agree 3. Neutral 4. Disagree 5. Strongly disagree 6. Not sure ii. It is user friendly 1. Strongly agree 2. Agree 3. Neutral 4. Disagree 5. Strongly disagree 6. Not sure In case you are not satisfied with the easiness of use, can you suggest how to improve it?
Incentive design component must provide information that allows the user to evaluate incentive designs	Policy Maker+Env. Expert	Please indicate which statement represents how you feel about the tool after you have used it. i. It allows to evaluate incentive designs 1. Strongly agree 2. Agree 3. Neutral 4. Disagree 5. Strongly disagree 6. Not sure
Incentive design component must be installable with reasonable effort	Policy Maker+Env. Expert	Please indicate which statement represents how you feel about the tool after you have used it. i. It is installable with reasonable effort 1. Strongly agree 2. Agree 3. Neutral 4. Disagree 5. Strongly disagree 6. Not sure

Problems occurred while using the component	Policy Maker+Env. Expert	Please describe any problems that occurred during your use of the tool (either technical or related to your understanding of its operation). i. Open text box
Adequacy of the service (Rate 1 to 10)	Policy Maker+Env. Expert	Please rate the adequacy of the tool in your opinion (1 = Meets no requirements, and 10 = Meets all requirements) i. 1-10 scale ii. Comment box

8.4 Opinion mining

Indicator	**Testers group**	**Questionnaire**
Software usability, in terms of being able to present the user with the correct and useful information that they are looking for	Policy Maker+Env. Expert	Please indicate which statement represents how you feel about the tool after you have used it. i. It is simple to use 1. Strongly agree 2. Agree 3. Neutral 4. Disagree 5. Strongly disagree 6. Not sure ii. It is user friendly 1. Strongly agree 2. Agree 3. Neutral 4. Disagree 5. Strongly disagree 6. Not sure In case you are not satisfied with the easiness of use, can you suggest how to improve it?
The opinion mining component should be easy to adapt to new domains	Developers	Please indicate which statement represents how you feel about the tool regards adapting it to new domains. i. It is simple to achieve that 1. Strongly agree 2. Agree 3. Neutral 4. Disagree 5. Strongly disagree 6. Not sure ii. It is very difficult to carry out the adaptation 1. Strongly agree 2. Agree 3. Neutral 4. Disagree 5. Strongly disagree 6. Not sure iii. In case you are not satisfied with the easiness of adaptation, can you suggest how to improve it?

The opinion mining component should facilitate extensions to the set of topics and/or web sites.	Developers	Please indicate which statement represents how you feel about the tool regards extending it to new topics and/or web sites. i. It is simple to achieve that 1. Strongly agree 2. Agree 3. Neutral 4. Disagree 5. Strongly disagree 6. Not sure ii. It is very difficult to carry out the extension 1. Strongly agree 2. Agree 3. Neutral 4. Disagree 5. Strongly disagree 6. Not sure iii. In case you are not satisfied with the easiness of extension, can you suggest how to improve it?
The opinion mining component should provide easy ways of exploring the tendency of the sentiment scores	Policy Maker Env. Expert	Please indicate which statement represents how you feel about the tool concerning exploring the tendency of the population sentiment. i. It is simple to use 1. Strongly agree 2. Agree 3. Neutral 4. Disagree 5. Strongly disagree 6. Not sure ii. It is user friendly 1. Strongly agree 2. Agree 3. Neutral 4. Disagree 5. Strongly disagree 6. Not sure iii. In case you are not satisfied with the easiness of use, can you suggest how to improve it?

The opinion mining component should provide means to drill down the scores to the actual messages that lead to the scores	Policy Maker+Env. Expert	Please indicate which statement represents how you feel about the tool concerning drilling down the sentiment scores up to the actual messages on e-participation tools i. It is simple to use 1. Strongly agree 2. Agree 3. Neutral 4. Disagree 5. Strongly disagree 6. Not sure ii. It is user friendly 1. Strongly agree 2. Agree 3. Neutral 4. Disagree 5. Strongly disagree 6. Not sure iii. In case you are not satisfied with the easiness of use, can you suggest how to improve it?
Ease of use	Policy Maker+Env. Expert	Please indicate which statement represents how you feel about the tool after you have used it. i. It is simple to use 1. Strongly agree 2. Agree 3. Neutral 4. Disagree 5. Strongly disagree 6. Not sure ii. It is user friendly 1. Strongly agree 2. Agree 3. Neutral 4. Disagree 5. Strongly disagree 6. Not sure In case you are not satisfied with the easiness of use, can you suggest how to improve it?
Problems occurred while using the component	Policy Maker+Env. Expert	Please describe any problems that occurred during your use of the tool (either technical or related to your understanding of its operation). i. Open text box
Adequacy of the service (Rate 1 to 10)	Policy Maker+Env. Expert	Please rate the adequacy of the tool in your opinion (1 = Meets no requirements, and 10 = Meets all requirements) i. 1-10 scale ii. Comment box

8.5 Visualization

Indicator	Testers group	Questionnaire
Visual encodings	Policy Maker Env. Expert Public user	In these questionnaires the users are asked whether they understand the visual designs and whether they intuitively know how to use the interfaces. i. Do you find the organization and display of the interface useful for your work? ii. Is any information provided that should be neglected in the interface?
Interaction designs	Policy Maker Env. Expert Public user	In these questionnaires the users are asked whether they understand the visual designs and whether they intuitively know how to use the interfaces. i. Looking at the visual interface – do you intuitively know, how to use the interface? ii. Do you have any suggestions for improving the interface?
Ease of use	Policy Maker Env. Expert Public user	Please indicate which statement represents how you feel about the tool after you have used it. i. It is simple to use 1. Strongly agree 2. Agree 3. Neutral 4. Disagree 5. Strongly disagree 6. Not sure ii. It is user friendly 1. Strongly agree 2. Agree 3. Neutral 4. Disagree 5. Strongly disagree 6. Not sure In case you are not satisfied with the easiness of use, can you suggest how to improve it?
Problems occurred while using the component	Policy Maker Env. Expert Public user	Please describe any problems that occurred during your use of the tool (either technical or related to your understanding of its operation). ii. Open text box
Adequacy of the service (Rate 1 to 10)	Policy Maker Env. Expert Public user	Please rate the adequacy of the tool in your opinion (1 = Meets no requirements, and 10 = Meets all requirements) iii. 1-10 scale iv. Comment box

8.6 Software architecture

Indicator	Testers group	Questionnaire
Accuracy, precision and re-configurability	Policy Maker+Env. Expert	Please indicate which statement represents how you feel about the tool after you have used it. i. It is accurate in the sense that it provides precise information about the plans and the impact that each plan has on the considered aspects (social, environmental, etc.)? 1. Strongly agree 2. Agree 3. Neutral 4. Disagree 5. Strongly disagree 6. Not sure In case you are not satisfied with the precision or accuracy, can you suggest how to improve it? i. It is easily configurable? 1. Strongly agree 2. Agree 3. Neutral 4. Disagree 5. Strongly disagree 6. Not sure in case you are not satisfied with the reconfigurability , can you suggest how to improve it?