Title:	Document Version:
D4.6 Data analytics for security policies	0.1

Project Number:	Project Acronym:	Project Title:
H2020-740466	LETSCROWD	Law Enforcement agencies human factor methods and Toolkit for the Security and protection of CROWDs in mass gatherings

Contractual Delivery Date:	Actual Delivery Date:	Deliverable Type*-Security*:
M19 (November 2018)	M20 (December 2018)	R-PU

*Type: P: Prototype; R: Report; D: Demonstrator; O: Other.

**Security Class: PU: Public; PP: Restricted to other programme participants (including the Commission); RE: Restricted to a group defined by the consortium (including the Commission); CO: Confidential, only for members of the consortium (including the Commission).

Responsible:	Organisation:	Contributing WP:
Fabio Roli	UNICA	WP4

Authors (organisation):
Giorgio Fumera (UNICA)

Abstract:
This document presents the results of task T4.2. Several data analysis and visualization functionalities are proposed for the user roles of the LETSCROWD application, which allow them to explore data about events and policies, and to extract information useful to carry out their event management tasks.

Keywords:
Policy Making Toolkit, Data analysis and visualization, Policy indicators

Revision History

Revision	Date	Description	Author (Organisation)
V0.1	20.12.2018	First draft	Giorgio Fumera (UNICA)

This project has received funding from the European Union's Horizon 2020 Research and Innovation Programme under Grant Agreement № 740466.

More information available at https://letscrowd.eu

Copyright Statement

Index

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

1.1 PURPOSE OF THE DOCUMENT

The Policy Making Toolkit (PMT) is a module of the LETSCROWD application that supports different user roles to manage an event through its phases, covering the tasks of data management, approval decision support, policy creation, and application of policies to events. It is based on event and policy data stored in the LETSCROWD application.

This document proposes some data analysis and visualization functionalities for the different LETSCROWD application user roles which would allow them to explore and visualize event and policy data for their specific needs, e.g., supporting policy makers to investigate and assessing the effectiveness of existing policies.

1.2 SCOPE OF THE DOCUMENT

This document describes the results of task 4.2 of work package WP4. In the first version (deliverable D4.2) *descriptive analytics* was identified as the most suitable kind of analytics functionality for the different end users of the LETSCROWD application, and the architecture of a *data analysis and visualization* module was sketched. Based on the results of task 4.1 (definition of security policy indicators, reported in deliverable D4.5) and on the first version of the Policy Making Toolkit (described in deliverable D4.4), in this document specific data analysis and functionalities are proposed for each user role, and their integration into the LETSCROWD application is discussed.

1.3 STRUCTURE OF THE DOCUMENT

This document is structured in five sections beside the introduction. Section 2 summarizes the event and policy information stored in the LETSCROWD application. In section 3 an overview of the LETSCROWD application user roles is given, and the possible data analysis tasks of interest for each role are sketched. Section 4 reports the main contribution of this document: it describes the data analysis and visualization functionalities which can be implemented in the LETSCROWD application, in the form of predefined kinds of queries to be provided to the users with a user-friendly, graphical representation of the results. Finally, section 5 briefly discusses implementation issues of the proposed functionalities.

2 EVENT AND POLICY DATA STORED IN THE LETSCROWD APPLICATION

2.1 EVENTS

Each event stored in the LETSCROWD application will be characterized by the values of a set of about 900 "policy indicators" defined in deliverable D4.5. Such indicators are subdivided into nine main categories: 1) Event information, 2) Crowd information, 3) Legal issues, 4) Public information, 5) Security organization, 6) Security information/intelligence, 7) Security planning, 8) Security measures/operations, 9) Post event actions. Each category of indicators is in turn hierarchically subdivided into subcategories, up to a depth of six.

The value of each indicator (at any depth) can belong to one out of six different types:

- Binary, e.g., true/false, or completed task/failed to complete task.
- Categorical; for instance, the type of an event can be sporting, cultural, religious, etc.
- Continuous, e.g., the density of a crowd (in a suitable unit of measurement).
- Discrete, i.e., an integer value; for instance, the number of exits of an event venue.
- Ordinal, i.e., a set of categorical values sorted according to some criterion; for instance, the size of a crowd can be categorized as small (< 1.000 people), medium (1.000 to 20.000 people) and large (> 20.000 people).

The values of the indicators for a given event are set by different users involved in event management (see section 3). Not all indicator values are mandatory: some of them can be left empty.

Beside the values of policy indicators, all the other event information generated by the different modules of the LETSCROWD application will be stored, e.g., information about incidents (if any) occurred during an event, weak signals generated and collected for dynamic risk assessment, etc. This information can be very useful for different user roles, as a basis for different kinds of analyses about past events and their relationship with policies.

2.2 POLICIES

Policies are defined in deliverable D4.4 as " statements of intent, implemented as procedures or protocols that can assist in subjective and objective decision making". In practice, a policy corresponds to a law or regulation in force in a given administrative entity (e.g., a country or a region) to manage a given kind of event.

The PMT allows certain user roles to define policies, and to select policies to be applied to an event in preparation (see section 3).

In the PMT a policy is made up of four components:[1]

- **Generic data**: name, description, and other reference information.
- A set of **conditions**: each condition sets a constraint on the value of one policy indicator, and the different conditions are combined with the logical operators AND and OR. If the conditions of a policy are not fulfilled for a given event, then such a policy cannot be applied to that event. As an example, the conditions of a policy may state that it can be applied to events where the crowd size is small and the estimated number of disabled people is at most five.
- A set of **mandatory** measures, which are evaluated only if the policy conditions (see above) are fulfilled. Similarly to conditions, each measure consists in a constraint on the value of one of the policy indicators; differently from conditions, *all* mandatory measures must be fulfilled, i.e., they are combined by a logical

[1] The structure of a policy described in deliverable D4.4 has been subsequently (slightly) changed: the one described here is the most recent structure at the time of writing this document.

AND. However, if some mandatory measures of a policy are not fulfilled for a given event in preparation (for which the conditions are fulfilled), such a policy is not excluded by the PMT: it is nevertheless taken into account, and is shown to the user with unfulfilled mandatory measures highlighted; the rationale is that it is assumed that the corresponding indicator values can be changed during event preparation to fulfil all mandatory conditions.

- A set of **actions** or **recommendations** set by the policy, that must be planned and later executed. They are written as free text.

It is also planned that a policies can be modified over time: in this case a new version of the same policy is created, and all previous versions are kept in the PMT database for reference. This can be a useful feature for carrying out several kinds of analyses of policies; for instance, an obvious application is to analyse the evolution of a given policy over time, possibly related with the effects of the different versions of that policy on the events they were applied to.

3 DATA ANALYSIS NEEDS OF DIFFERENT USER ROLES

This section discusses the specific data analysis needs that can be envisaged for the different LETSCROWD application user roles planned in deliverable D4.4. To make this document self-contained, an overview of such user roles is first given in section 3.1, and a concise description of each of them is given in the corresponding subsequent sections.

3.1 PLANNED USER ROLES

According to deliverable D4.4 the PMT contains different user roles, and each role is given access only to relevant data. In particular:

- a given role may have access to certain events and policies, but not others;
- even if a given role can access a certain event, some event indicators can be not visible to it;
- if a role can access a policy, all the policy components are visible to it.

The following roles are currently defined: **administrator**, **organizer**, **event authorizer**, **LEA**, **policy maker**. It is also planned that new roles can be defined using existing ones as a base; for instance, several country-specific LEA roles can be defined, and each of them will have limited access to certain data, e.g., only to events which took place in the respective country.

Finally, it is worth noting that a given user of the LETSCROWD application can have more than one role.

In the following subsections each the above mentioned roles will be discussed to identify the kinds of data analysis and visualization functionalities of potential interest to them.

3.2 ADMINISTRATOR

This is a special role which can manage the other roles and create new ones. In deliverable D4.4 the specific definition of the functionalities of this role was postponed to the second version of the same deliverable. At this time no specific data analysis and visualization functionalities can be planned for this role.

3.3 ORGANIZER

The organizer is the entity (person or company) that creates and organizes an event. With respect to the LETSCROWD application, an organizer can only access events it created, and only if the event state is "Preparation" (see the event lifecycle in Figure 1); moreover, only a subset of the event indicators can be accessed by this role. Finally, it is planned that organizers will carry out this task using a distinct application from the PMT.

If organizers are given access only to their own events in preparation, no specific analysis and visualization functionalities are needed for this role. On the other hand, if organizers were also given access at least to *past* events they created (i.e., events whose state is "Finished" – see Figure 1), then they could be

interested in searching and retrieving data about such events for further analysis; for instance, they may want to compute some statistics about events organized so far, although limited to the event indicators they are allowed to access.

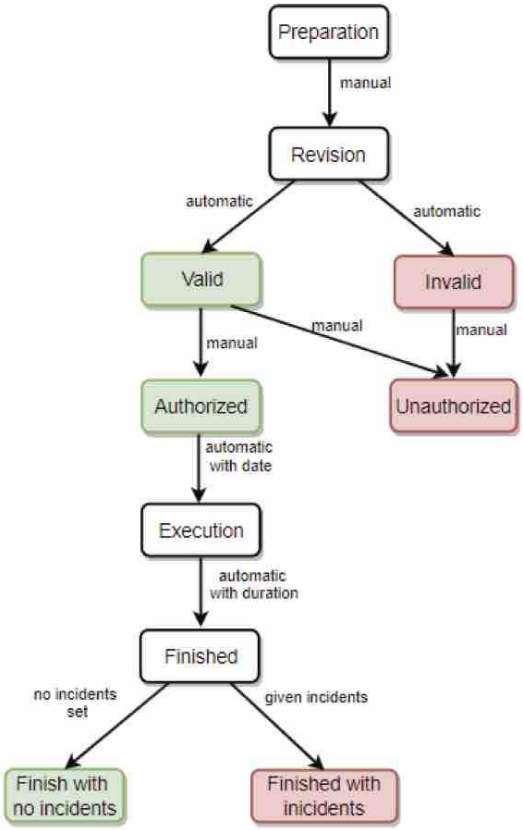

Figure 1 – Event states (taken from deliverable D4.4).

3.4 EVENT AUTHORIZER

The event authorizer is involved in the pre-event phase ("Preparation" and "Revision" states): it moves an event from the "Preparation" to the “Revision” state after all the necessary information has been provided by the organizer, and will then move it to the "Authorized" or "Unauthorized" state. The event authorizer can also create and edit events, and has access to the majority of the event indicators. Finally, this role apply existing policies, and create reports for events.

The above tasks can be supported by a richer set of data analysis and visualization functionalities involving both events and policies. In particular, the event authorizer role can be seen as a potential source of useful information or feedback for policy makers (see section 3.6) about the suitability or effectiveness of policies to specific kinds of events; under this viewpoint this can be seen as a complementary role to the LEA role (see section 3.5).

3.5 LEA

This is a special role in the LETSCROWD application, as LEAs are the main end users of the project outcomes and can access different modules of the whole application. It is planned that every specific LEA will be assigned a custom LEA role. The definition of the exact functionalities of this role was postponed to the second version of deliverable D4.4; nevertheless, it can be envisaged that each specific LEA is given access to the information of at least all the events it was involved in (possibly, all the event indicators), and possibly some information (at least a subset of the event indicators) about events in the same country if managed by a different LEA, e.g., as in the case of Madrid and Bilbao police forces in Spain.

It may be useful for LEAs also to access information about existing policies, at least the ones in force in their own country. In particular, LEAs are likely to be a source of very useful information and feedback for policy

makers about the effectiveness of existing policies, based on their first-hand experience in the field; for instance, they may provide suggestions about modifications to specific policies to improve their effectiveness or even their applicability, as well as suggestions about new possible policies. Accordingly, the LETSCROWD application should provide specific data analysis and visualization functionalities to support LEAs also in this task.

3.6 POLICY MAKER

The policy maker role refers to the entity in charge of creating and managing policies (i.e., editing the corresponding fields in the PMT). It is also planned that this role has access to event information, which is useful to the policy creation process (e.g., taking into account incidents occurred during past events), but cannot edit events. Policy makers will be also given the possibility to create reports about existing policies.

Given the high-level profile of this role, it can be assumed that it will be able to access all the information about policies stored in the LETSCROWD application, possibly including policies in force in different countries, as well as all or most of the information about past events, at least the ones in the same country.

3.7 OTHER ROLES

It is planned that additional roles can be created by the administrator user, as more specific instances of one of the roles discussed above. For instance, each country-specific LEA will be assigned a specific LEA role distinct from the one of other LEAs, and therefore characterized by different access permissions to data stored in the LETSCROWD application. Further details are going to be defined in the second version of deliverable D4.4.

The data analysis and visualization functionalities described in this document for each main role are assumed to be "inherited" by more specific instances of the same role, with possible additional limitations when appropriate, e.g., limiting access to data related to a specific country.

4 DATA ANALYSIS AND VISUALIZATION FUNCTIONALITIES

In this section nine different kinds of data analysis and visualization functionalities for the planned user roles of the LETSCROWD application are proposed. They consist in:

- allowing users to make specific kinds of queries to retrieve data about the two main categories of information stored in the LETSCROWD application: events and policies;
- carrying out some analyses on the retrieved data (only some of the proposed functionalities);
- providing an easy-to-understand graphical representation of the retrieved data and of the outcome of their analysis (if any), and allowing users to access further related information.

Such functionalities are intended to provide information which can provide useful insights to the users, helping them in their event or policy management tasks, as well as enabling further analyses.

4.1 RETRIEVING EVENTS

This functionality consists in retrieving all the events stored in the LETSCROWD application that can be accessed by a given user role. The output consists first in graphically showing the name and location (and possibly the date) of the retrieved events on a map.

Events may be shown in different colours according to their state (see Figure 1); only three states may be considered for the purposes of this functionality: Preparation (which would include Revision, Valid and Authorized), Execution and Finished.[2]

[2] Some user roles may have access only to events in certain states, e.g., only to events that already took place ("Finished"). In such a case, only events in the corresponding states would be shown.

Users should then be allowed to view further information about a given event (limited to the data that can be accessed by the corresponding user role) by clicking on its graphical representation on the map. Event information should be shown either in a pop-up window, or in a fixed, dedicated area of the window, or in a separate window. In any case, it should be shown in a hierarchical way using a multi-level structure with higher-level elements that can be expanded to show lower level ones. For instance, the highest level could show only labels characterizing the available information, e.g., policy indicators, applied policies, incidents (only for events which already took place), and risk information; each label can then be selected to access lower-level details (e.g., policy indicators can be accessed according to their hierarchy).

Users should also be allowed to select a subset of the retrieved events through a filtering functionality which should provide at least the following filtering options:

- date (i.e., selecting only events on a certain date or in an interval of dates);
- location (e.g., country, region, city);
- event type, according to the values of the corresponding policy indicator;
- crowd size, according to the values of the corresponding policy indicator.

The result of the filtering operation is to show on the same map only the events satisfying the chosen conditions.

Users should also be allowed to select events to be used in some of the following functionalities, for instance to compare two or more events (see section 4.3) and to search for events similar to a given one (see section 4.4).

Finally, the user should be given the possibility to save information about retrieved events (or a subset of them), at least in two ways:

- exporting the map show on the screen in a graphical format;
- exporting information about selected events (only the ones which can be accessed by the user role), or a subset of that information, to a report in text or PDF format.

4.2 SEARCHING FOR EVENTS

This functionality allows users to search for events exhibiting specific characteristics defined in terms of values of indicators of interest (e.g., event type, size of the crowd, type of the venue, etc.), and on other information like the occurrence of incidents or of specific weak signals, provided that the corresponding kind of information can be accessed by a given user role.

This functionality should be provided as a search mask where the user can select the information of interest (e.g., the "crowd size" indicator), set the corresponding value (e.g., "crowd size" = small), and start the search process. The user should be given the possibility of combining the chosen search conditions using the Boolean operators AND, OR and NOT (e.g., "crowd size" = small AND "event type" = sporting). With respect to the "filter" option of the "Retrieving events" functionality (see section 4.1), this one should allow users to set conditions on all the event information which can be accessed by the corresponding roles.

Events fulfilling the query conditions should be graphically shown in a map as described in the previous section.

The graphical interface should also give users the possibility to refine a query by adding, removing or modifying conditions.

Users should also be allowed to export results as described in the previous section.

4.3 COMPARING EVENTS

A functionality which is likely to be of interest to several user roles is to provide a comparison between two or more events, to highlight similarities and differences between them. For instance, this may allow to analyse the possible reasons that one event took place without any problem, whereas another event exhibited several problems (e.g., organizational problems, incidents, etc.).

This functionality should allow the user to select the events of interest (for instance, from a map) and should then provide a graphical representation of their main characteristics (indicator values, and other event information), in such a way that similarities and differences between them become as evident as possible by a quick visual inspection. Only a subset of event information is likely to be easily visualized in this way, for instance indicators like the crowd size and event type can be visualized using different colours, geometrical shapes of different size, etc.

The user should then be allowed to inspect other event information by accessing it as explained in section 4.1.

4.4 FINDING SIMILAR EVENTS

The functionalities defined in the two previous sections can be seen as standard database querying tasks. A different kind of data analysis functionality of potential interest for different user roles is the search for events "similar" to a given one. Here the concept of "similarity" is not rigorously defined, e.g., in terms of the fulfilment of a Boolean condition on the values of some policy indicators, but is instead only qualitative.

For instance, an event authorizer who receives a request for a new event in preparation may want to analyse previous "similar" events that have already taken place, before taking a decision, e.g., to see whether incidents occurred in such events, or to see what policies had been applied to them events, if different policies are applicable. As another example, a policy maker may want to analyse policies applied to similar events, e.g., to evaluate their effects.

The similarity criterion should be defined in terms of the available event information, i.e., the values of the policy indicators, the incidents occurred (if any, and only if also the event at hand has already taken place), the applied policies (also in this case, only if the event at hand has already taken place), and possibly information related to risk like weak signals. General specifications on the similarity criterion are the following ones:

- predefined similarity criteria should be used, instead of letting users to define them from scratch;
- indicators which have not been assigned a value should not be considered, as well as information that cannot be accessed by the given user role (otherwise this functionality may reveal some information which should not be accessed);
- users should be allowed to indicate specific event information they would like to focus on when searching for "similar" events (e.g., event type and crowd size), but not to specify values of the corresponding indicators nor Boolean combinations of conditions on such indicators, since this would correspond to the search functionality of section 4.2;
- users may be allowed to specify the relevance that each component of event information should have in the evaluation of similarity; relevance should not be expressed in a numeric scale, which would be somewhat arbitrary and not clearly interpretable, but rather qualitatively, e.g., using a small set of discrete labels like LOW, MEDIUM, HIGH; anyway this possibility should be limited to the main categories of policy indicators;
- even if the user does not specify any kind of event information to focus on, it may be not convenient to use all the available event information to evaluate the similarity (taking into account that about 900 policy indicators have been defined): also in this case the similarity criterion should be based only on a suitable subset of event information;

- after the "similar" events have been retrieved (see below) the user should be given the possibility to refine the results by changing either the subset of event information to be used to evaluate similarity or the relevance of the different components.

Taking into account the above requirements, a possible similarity measure could be defined following the approach of content-based information retrieval systems (1), which allow users to retrieve data (e.g., documents or images) similar to an example provided by the same user. To this aim, for each policy (1)indicator which is planned to be involved in the evaluation of event similarity a suitable similarity measure among its values should be defined, according to the type of such values (see section 2.1); in particular:

- For a numerical indicator X, either continuous or discrete (e.g., the density of a crowd in a suitable unit of measurement): the similarity function between the values $X_1$and X_2 it takes in two events can be defined as a decreasing function of the absolute difference $|X_1 - X_2|$.
- For ordinal (discrete but not numerical) indicators an analogous similarity function can be defined, in terms of an ordinal (discrete) distance between the values $X_1$and X_2.
- For categorical indicators whose values are not ordered (e.g., the type of an event can be sporting, cultural, religious, etc.) no general similarity function can be defined: the simplest solution is to use a binary measure (similar/dissimilar), by analysing each pair of values and defining which ones can be considered "similar" and which ones "dissimilar".
- For binary indicators the only possible choice is to consider "similar" a pair of identical values and "dissimilar" a pair of different values.

All the above similarity functions should output a value in the range $[0,1]$ (where 0 denotes the lowest similarity, and 1 the highest similarity). In the particular case of binary functions only the extreme values 0 and 1 should be used. The overall similarity between two events can be computed as a value in the same range $[0,1]$ by combining the similarity values among the considered components (indicators), weighting them with the qualitative relevance indicated by the user (if any). To this aim, the relevance values LOW, MEDIUM and HIGH could be converted into numerical weights, e.g., respectively 0.1, 0.5 and 1.0. In this case all the indicators whose relevance has not been indicated by the user should be assigned a weight of 1.0.

After the similarity score with the event at hand has been computed for all available events, such events should be presented to the user sorted according to their score (from highest to lowest). In practice only the top *K* events in the list can be shown for a suitable value of *K*, and further events can be shown only upon user request. Similarly to the first two functionalities above (sections 4.1 and 4.2), retrieved events can be shown on a map; in this case different gradations of a same colour could be used to visually represent the similarity score.

Obviously users should be allowed to access information about retrieved events (limited by the permissions associated to the user roles), which should be shown together with the corresponding information of the event selected to start the search process.

4.5 CLUSTERING EVENTS

Another functionality based on a qualitative evaluation of the "similarity" between events is to subdivide past events into several groups. This functionality may be useful to policy makers and to LEAs, to give them a high-level overview of past events, and to highlight groups of events exhibiting similar characteristics. For instance, it may help detecting whether the same kind of incidents tend to occur in certain kinds of events, what policies have been applied to similar events, etc.

To this aim the same similarity criterion discussed in section 4.3 can be used. The same kind of user input can be considered as well, with the only exception that in this case no initial event is selected. More precisely, a pre-defined similarity measure is considered, and the user is allowed to select a subset of event

information (among the information that can be accessed by that user) and to specify the relevance of its different components in the evaluation of the similarity between events.

The automatic detection of distinct groups of events based on the above similarity criterion can then be achieved through suitable *clustering* algorithms (2) (3), chosen among the ones developed in the data mining field.

The results can be shown to the user using ad hoc graphical representations which show the clustered entities (in this case, events) as points in a two- or three-dimensional space, such that the geometric distance among every pair of points is as much as possible proportional to the (dis)similarity between the corresponding entities evaluated using the underlying similarity score (4). As a simpler alternative, events can also be shown in a map using different colours, one colour per cluster, provided that the number of clusters is small enough (say, up to ten) to allow users easily distinguish the different colours.

A more basic clustering functionality can be implemented, which consists in grouping events on the basis of a simple criterion specified by the user as a condition on the values of a single policy indicator or of other event information. This can be useful if the user is interested in identifying distinct groups of events in terms of a given feature of interest, e.g., grouping events on the basis of the crowd size. Accordingly, the condition on the values of the information of interest is defined by subdividing the range of such values into a certain number of intervals (in case of continuous indicators) or into subsets (in case of discrete indicators).

The outcome of the above, basic clustering process can be shown to the user using the same graphical representations discussed above.

4.6 RETRIEVING POLICIES

This functionality is analogous to the one described in section 4.1 for events. It consists in retrieving all the policies stored in the PMT that can be accessed by a given user role. This is very likely to be of interest for policy makers, but can be useful also to event authorizers (limited to the policies they can access), as well as to LEAs.

Retrieved policies should be shown not as a bare list, but grouped according to some meaningful criteria (possibly arranged into a hierarchy), like the country where they are in force, the administrative level they refer to (national, regional, etc.), the kind of event they regulate, etc. An alternative is to graphically show the policies in a map according to the reference country. In both cases, users should be allowed to easily access three main kinds of information about each policy (or a subset of them, depending on user role permissions):

- Policy definition: description, conditions, mandatory measures, and actions (see section 2.2).
- Previous versions of that policy (if any). Different versions of the same policy could be graphically shown in such a way to allow users to easily spot similarities and differences, e.g., in a sort of timeline.
- Past events to which that policy was applied, including its previous versions; such events can in turn be graphically represented in a map (possibly on the same map where policies are shown) as described in section 4.1.

A specific information that should be easily accessible by the user when accessing the retrieved policies is related to the incidents occurred in events which a policy was applied to. The most basic information is whether incidents occurred in any such event, and how many events resulted in incidents. Such events should in turn be easily accessible to the user for further inspection.

4.7 SEARCHING FOR POLICIES

This functionality allows a user to search for policies that satisfy a given query defined by the user itself. Two main kinds of search criteria should be considered:

- Constraints on the values of the different components of the policy: description, conditions, mandatory measures and actions. In particular (see section 2.2 for the structure of a policy):
    - the description and the actions of a policy are represented as free text fields; in this case the query should be a textual search, e.g., one or more words that should be present in the description or action components;
    - conditions and mandatory measures are expressed in terms of constraints on the values of policy indicators; in this case the query should allow users to select policies whose conditions or mandatory measures refer to specific indicators.
- Constraints on the events to which policies of interest have been applied. These constraints should refer at least to:
    - the main event information (event type, crowd size, etc.);
    - the occurrence of incidents in events where the policies have been applied.

The query can be expressed by combining two or more constraints using the Boolean operators AND, OR and NOT. The graphical interface should give users the possibility to refine a query by adding, removing or modifying constraints.

Similarly to the "Searching for events" functionality of section 4.2, also this functionality should be provided through a search mask where the user can express the query and then start the search process. Retrieved policies can be shown to the user using the same graphical representation described in section 4.6.

4.8 COMPARING POLICIES

This functionality allows users to compare two or more policies, including different versions of the same policy. This can be especially useful to policy makers, for instance to analyse the evolution of a given policy over time, or to see how a given kind of event is regulated in different countries.

To this aim the graphical interface should first allow users to select the policies to be compared; this should be possible also from the outcome of the two functionalities described above (retrieving all policies and searching for specific policies). Then a graphical representation of the main information about such policies should be provided, highlighting similarities and differences; as an example, policy conditions or mandatory measures referring to the same indicators could be highlighted with identical colours.

To ease the graphical representation of similarities and differences not all the policy components should be immediately shown, but a hierarchical representation should be used, allowing the user to deepen the level of comparison only to components of interest. For instance, only the names of the indicators involved in the conditions and mandatory measures could be initially shown, whereas the constraints on the corresponding values and their Boolean combination could be shown only upon user request.

4.9 FINDING SIMILAR POLICIES

Analogously to the "Finding similar events" functionality of section 4.4, this one allows users to search for "similar" policies to a given policy of interest. The most likely user roles interested in this kind of search are the policy maker and event authorizer ones.

For instance, a policy maker who is analysing a policy from its own country may want to see whether similar policies exist in different countries. Analogously, an event authorizer who is going to apply a local policy (e.g., defined at a regional level) to an event in preparation may be interested in checking whether similar policies exist in other regions of the same country.

In the case of policies the similarity criteria can be defined according to their four components:

- textual similarity between their description and between their actions (both represented as free text), which can be evaluated in terms of the occurrence of the same or similar (synonymous) terms;

- degree of overlapping between the sets of policy indicators involved in the respective conditions and mandatory parts (e.g., fraction of common indicators among the set of all indicators involved in the conditions, and the same for mandatory actions);
- degree of "compatibility" between the constraints set in the conditions and in the mandatory actions to the values of common indicators (e.g., two policies including a condition stating that the crowd size is small, vs one policy referring to small crowd size and another policy referring to large crowd size).

For each pair of policies under comparison, the similarity between each of their four components should be separately evaluated through a similarity score in a given range (e.g., [0,1]); a final similarity score should then be computed by combining the above four scores, possibly allowing user to specify the relevance of each of the four components by assigning qualitative weights to them, e.g., in a scale LOW, MEDIUM, HIGH as suggested for events (see section 4.4). The user should also be allowed to specify only a subset of components to be used in the similarity evaluation.

The results can be presented to the user in a similar way as for the search of similar events. After the similarity scores between all the available policies and the one at hand have been computed, such policies should be presented to the user sorted according to their score (from highest to lowest), possibly shown on a map (e.g., according to the country or region where they are in force) using different colour gradations to represent the similarity value; users should then be allowed to access information about the retrieved policies, which should be shown together with the corresponding information about the policy used to start the search process.

4.10 SUMMARY

Based on the analysis of user roles in section 3, the following functionalities proposed in section 4 can be made available to the different user roles:

- The **Administrator** user role should be not interested in using the proposed functionalities.
- The **Organizer** role should be given access at least to the first four functionalities involving events (sections 4.1–4.4), whereas functionalities related to policies are not likely to be of interest to it, as well as the more advanced clustering functionality for events.
- The **Event authorizer** role should access at least to the first four functionalities involving events and the first three ones involving policies (sections 4.6–4.8). The more advanced functionality of each category (clustering events and finding similar policies) could be excluded for this role.
- The **LEA** and **Policy maker** user roles should be given access to all the proposed functionalities.

The above mapping from user roles to functionalities is also summarized in Table . As already pointed out in the previous sections, each user role could have limited access permissions to the information about event and policies.

	User roles				
Functionalities	**Administrator**	**Organizer**	**Event authorizer**	**LEA**	**Policy maker**
Retrieving events		X	X	X	X
Searching for events		X	X	X	X
Comparing events		X	X	X	X
Finding similar events		X	X	X	X
Clustering events				X	X
Retrieving policies			X	X	X
Searching for policies			X	X	X
Comparing policies			X	X	X
Finding similar policies				X	X

Table 1 – Data analysis functionalities suggested for each user role.

5 IMPLEMENTATION AND INTEGRATION WITH THE LETSCROWD APPLICATION

As described in deliverable D4.4, all the data of the LETSCROWD application (including event and policy data) will be stored into a centralized document-oriented database which will be accessible by the different application modules.

The functionalities proposed in this document involve three kinds of processing steps:

- retrieving data about events or policies;
- processing retrieved data (only for some functionalities);
- displaying in a graphical format the retrieved data and the results (if any) of the processing tasks, using mainly map representations.

For all the proposed functionalities data retrieval can be implemented using the querying functionalities of the LETSCROWD application database. Data processing tasks (when required) involve only basic operations on the retrieved data: this can be implemented by software tools using any programming language, and easily integrated with the LETSCROWD application. Visualization of the results requires a map displaying tool, which is already available in the current version of the LETSCROWD application.

6 REFERENCES AND ACRONYMS

6.1 REFERENCES

1. **van Rijsbergen, C. J.** *Information Retrieval.* s.l. : Butterworth, 1979.

2. **Jain, Anil K. and Dubes, Richard C.** *Algorithms for Clustering Data.* s.l. : Prentice-Hall, 1988.

3. **Jain, Anil K.** *Data clustering: 50 years beyond K-means.* Pattern Recognition Letters, Vol. 31, No. 8, 2010, pp. 651-666.

4. **Ioannidis, Y.** Visual Representation. [book auth.] L. Liu and M. T. Oezsu. *Encyclopedia of Database Systems.* Boston: Springer, 2009.

6.2 ACRONYMS

Acronyms List	
LEA	Law Enforcement Agency
PMT	Policy Making Toolkit

TABLE 2 – List of acronyms.