# *Scrivania*: Public services execution and Semantic Search

Riccardo Cognini, Damiano Falcioni, Marco Maccari, Alberto Polzonetti and Barbara Re
Computer Science Division
University of Camerino
Camerino (MC), 62032, ITALY
{name.surname}@unicam.it

*Abstract*— **Public Administration sector in modern society is characterized by the need to support extremely complex processes in order to provide services to citizen. In this area, transparency is one of the most important requirement to improve the quality of provided public services. In this paper we present a tool supporting PAs collaboration and transparency, named Scrivania. It allows Public Administration employee to model and publish their services and then be guided during the Business Processes model execution. Instead, citizens, using Scrivania, can search and execute the provided services, tracing their execution and, in case of delay, observe the state it occurs. Citizens can also be helped by a semantic search in order to execute specific tasks in a process.**

*Keywords—Public Administration, BPMN, Collaborative Modeling, Guided Execution, Semantic Categorization*

## I. INTRODUCTION

Public Administration (PA) sector in modern society is characterized by the need to support extremely complex processes in order to provide services to citizens. Complexity is due to the fact that services provisioning is commonly a collaborative activity shared among different, possibly many, PA offices. PA services are in general interrelated, so the enhancement of activities involved in a service delivery can generally impact on other activities. In this domain, transparency is one of the most important requirement to improve administration efficiency and citizens satisfaction [1]. With transparency we mean the ability for the PA to make citizens aware of the delivery process in terms of activities and people involved in its execution and governance, improving the citizen's perceived trust. A service supports transparency if for each task, citizen knows exactly how to perform it and in case of delay know where is the bottleneck.

In order to support such scenario we have designed and developed a tool named *Scrivania* that implement transparency into PA and improve citizen's satisfaction about the e-government services usage. In this paper we focalize on the *Scrivania* use cases and how they are implemented in the collaborative BP environment that allow PAs to collaborate during the modeling phase of a service. This environment provides a real time collaborative BPMN 2.0 modeler that PA managers can use in order to design and provide PA services. Citizen can execute this services using the *Scrivania* services execution environment that integrate a semantic categorization

engine. The engine is based on the state of the art in the related areas. It has been trained using English and Italian corpus documents and make use of a specific domain ontology in order to allow the correct categorization of common PA data and documents. Using this categorization is possible to improve the search and the collaborative functionalities offered by *Scrivania*, applying a semantic stamp.

The rest of the paper is organized as follows. Section 2 presents a background in BP modeling. In Section 3 the Italian e-gov scenario is presented. Section 4 presents *Scrivania* use cases. Then, Section 5 presents the architecture of *Scrivania*, services collaborative functionalities and semantic search during the execution of services. Finally, in Section 6 some conclusions are reported.

## II. BACKGROUND ON BP MODELING

We refer to a BP as "a collection of related and structured activities undertaken by one or more organizations in order to pursue some particular goal. Within an organization, a BP results in the provisioning of services or in the production of goods for internal or external stakeholders. Moreover BPs are often interrelated since the execution of a BP often results in the activation of related BPs within the same or other organizations" [2]. BPM supports BP experts providing methods, techniques, and software to model, implement, execute and optimize BPs which involve humans, software applications, documents and other sources of information [3]. Recent work has shown that BP modeling has been identified as a fundamental phase in BPM. The quality of BPs resulting from the BP modeling phase is critical for the success of an organization. However, modeling BPs is a time-consuming and error-prone activity. Techniques which can help organizations to implement high-quality BPs, and to increase process modeling efficiency, has become an highly attractive topic both for industries and for the academy. Certainly many different commercial tools have been developed to support BPM. Nevertheless for what concerns the modeling phase they mainly provide support for BP editing and syntactical analysis. Different classes of languages to express BPs have been investigated and defined. There are general purpose and standardized languages, such as the BPMN 2.0, the UML Activity diagram, or the Event-Driven Process Chain and many others. There are also more academic related languages, like the Yet Another Work-flow Language based on Petri Nets

that's the most prominent example. Among the listed languages there are several differences. These are related to the level of rigor, going from semi-formal, with a precise syntax and with semantic given in natural language, to formal languages for which the semantic is provided thanks to well founded mathematical theories.

In our work we refer to BPMN 2.0 [5], an Object Management Group (OMG) standard. This is certainly the most used language in practical context. The standard specifies three views: process, choreography and collaboration. The process view refers to private and public BP. Using a private BP model intra-organization Business Processes are represented. At the same time in public BP the interactions between a private Business Process and another BP or Participant is modeled. The collaboration diagram, is used in order to have a complete representation both of internal process as well as of the message exchange structure. BPMN 2.0 collaboration given an intuitive graphical notation reported in Figure 1. In particular, the following BPMN 2.0 collaboration elements are the most commonly used in this diagram specification.

- Pools and lanes are used to represent a participant or an organization involved in the BP, and they contain the private BP and related elements as reported in the following.

- Tasks are used to represent an action to perform that produces a result. Different types of Tasks exist, just to cite a few we refer to manual, service, human, etc. Tasks are graphically drawn as rectangles with rounded corners.

- Events are used to represent something that can happen. In particular, Start events represent the points in which the BP starts, intermediate events represent something that can happen during the BP execution, like time exceeding a deadline or the reception of a message, and End events are raised when the BP terminates. Different types of events can be introduced starting from this three main categories. Events are graphically drawn as circles.

- Gateways are used to manage the process flow on choices and parallel activities. Different types of gateways are available, the most used are exclusive and parallel. An exclusive gateway gives the possibility to describe choices in the BP and a single output path can be activated each time the gateway is reached. Parallel gateway have to wait all their input flows to start and then all the output paths are started in parallel. Gateways are graphically drawn as diamonds.

Finally, choreography defines the expected behavior between interacting participants. In particular, the following BPMN 2.0 choreography elements are the same of collaboration except for task that includes information about participants. Choreography Tasks are used to represent the communication between two or more participants. They are graphically drawn as rectangles with rounded corners. Inside choreography task there are participants; the sender is drawn as a white rectangles and the receiver is drawn as a gray rectangles. They can be drawn upward or below in any order. Besides pools and lanes are not considered in BPMN 2.0 choreography because participant information are included in tasks.
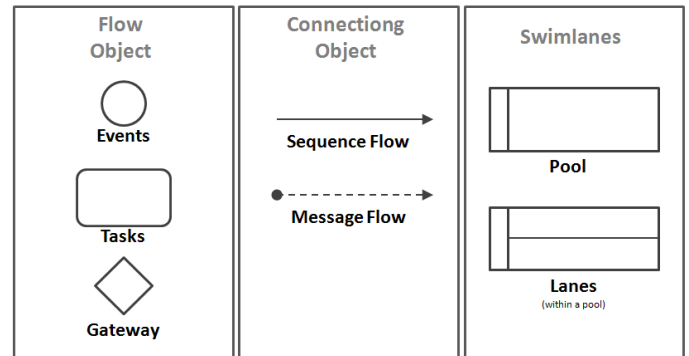


Figure 1. Example of a figure caption.

III.    ITALIAN SCENARIO ANALYSIS AND THE APPROACH

The e-government in Italy is characterized by services availability but low service usage. This is common in small towns that represent 70.3% of all National Cities. From the point of view of the citizen there is an emerging demand for updated information and fully interactive public services. The citizens want to be aware about the to do list before using the service through the Internet. This is commonly done, asking friends or directly contacting the PA. This can all be traced in order to understand the level of citizen satisfaction and consequently review the administration process.

Moreover, e-government services cannot be based only on a single PA, but there are scenario characterized by complex inter-administration processes, where each PA contributes in relation of its responsibilities. In such a way, Public Administration communities can emerge, following the success story of popular social networks.

The proposed approach aims to improve the experience of citizens interacting with PA. With Scrivania we want ensure to PA a tool for continuous improvement.

The proposed scenario raises multidisciplinary issues that can be solved at technological level using Knowledge Management (KM) [6] and Business Process Management (BPM) [4]. Talk about KM and BPM in Public Administration means to have knowledge-intensive and collaborative processes. Scrivania address KM and BPM in two main areas: respectively (i) the citizens profiling and (ii) the service modeling.

First, in order to return to the citizen information and services more attractive than their own interests, Scrivania allow an accurate profiling of the individual. It can be implemented either with explicit and implicit mechanisms. The explicit profiling is done via the interpretation of bookmarks or through analysis of the registration data. The implicit profiling is done to taking into account several indicators such as the time spent on a page and the browser history.

Studies in Public Administration underscore the ongoing transformation on the service life cycle. The traditional processes in the public sector, historically driven by internal goals, are changing in a outside provision. That's mainly thanks to the evolving of new technologies.

The conventional model of PA who works as a separate and distinct entity, managing its own knowledge disconnected from each other, is being transformed. The new scenario is characterized by network relations in whom governments need to cooperate with other governments, with non-profit organizations, businesses and citizens to deal with the new challenges of globalization. New actors and stakeholders enter in relationship with the government. In this context it is essential to support administrative processes. Summing up, the transformation of PA service models, can be contextualized with respect to the following points as key factors of success: collaboration, control, sharing, transparency, inclusiveness and simplification.

## IV. ACTORS AND USE CASES

The main actors of *Scrivania* are PAs employees and citizens. The most interesting use case related these actors are modeled using UML 2.0 as Figure 2 and 3 show. They provide a graphical representation of actor's interaction with *Scrivania*. Use cases can be divided in two groups:

- *Scrivania* services use cases are related to services creation and execution;

- *Scrivania* social environment use cases that are related to citizens social activities and interaction between actors.

In particular, *Scrivania* services use cases refer both to citizens and PA employs playing the role of administrator. As Figure 2 shows, citizens can search services, execute a selected service, check its state and evaluate the quality of the provided service. Instead, PA employs can create new services and make them available to citizens, manage services (for example in the case better versions of services are available) and check statistical indexes about services and citizen's opinions about tool and provided services.

*Scrivania* social environment use cases refer mainly to citizen as Figure 3 shows. They can search friends and send friendship requests. In this way they can communicate during a service execution if they want. Citizen can comment an executed service or an executed activity in a service, they can also read comments of friends related to services or activity. PAs and citizens employs a can be driven during service execution reading other citizen opinions.

## V. THE TOOL

### A. Overview on Scrivania architecture

*Scrivania* is implemented as a web based application running on Apache Tomcat. It permits to use java code directly to dynamic web pages. Thanks to jQuery we have implemented most of supported functionalities, for example the collaboration environment is created using jQuery post functions and java

code. The used Database Management System is MySQL. It contains all the needed information such as citizens and administrators data, services models, services instances, etc. Then, *Scrivania* can be installed in each operative system because all the used technologies are free and open source.

Being a web application, PAs and citizens can use *Scrivania* via web browser (Internet Explorer, Firefox, Safari, etc).It means that *Scrivania* is a 3-tier application where the complexity of client is managed in the server.
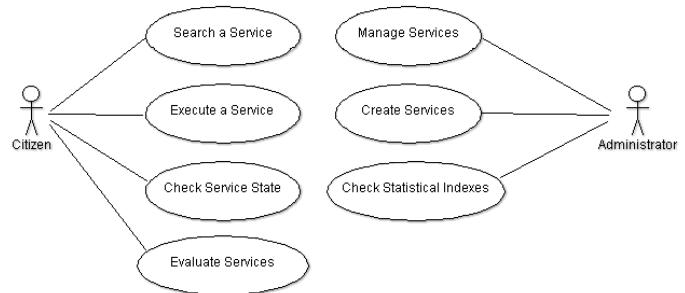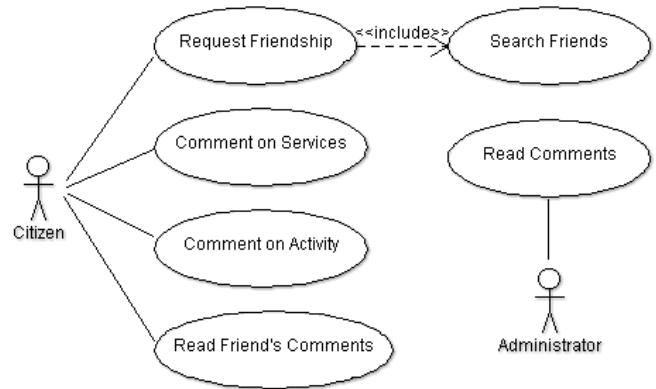


Figure 2. *Scrivania* Services Use Cases.



Figure 3. *Scrivania* Social Environment Use Cases.

The *Scrivania* architecture is based on a web application, suitable to support scalability and modularity. The main components are showed in Figure 4 and following described:

- *Authentication Module*, It allows citizens and administrators access to the functionalities provided by *Scrivania*. It guaranties three different levels of security that are following described. *Level 0* (self-registration) where users can register themselves and use such credential with a low level of security. *Level 1* (user name and password) is the most common and simple authentication system to administrate. It offers a lot of advantages, for example it does not need special hardware devices but it also presents the disadvantage that the association between the identity of people and authentication data is not guaranteed. Typically, this method is used to trace the activity of the user (profiling) and it grants a low protection level services access. Finally, *Level 2* (smart cards) is an authentication system based on physical support that guarantee the association between real identity and
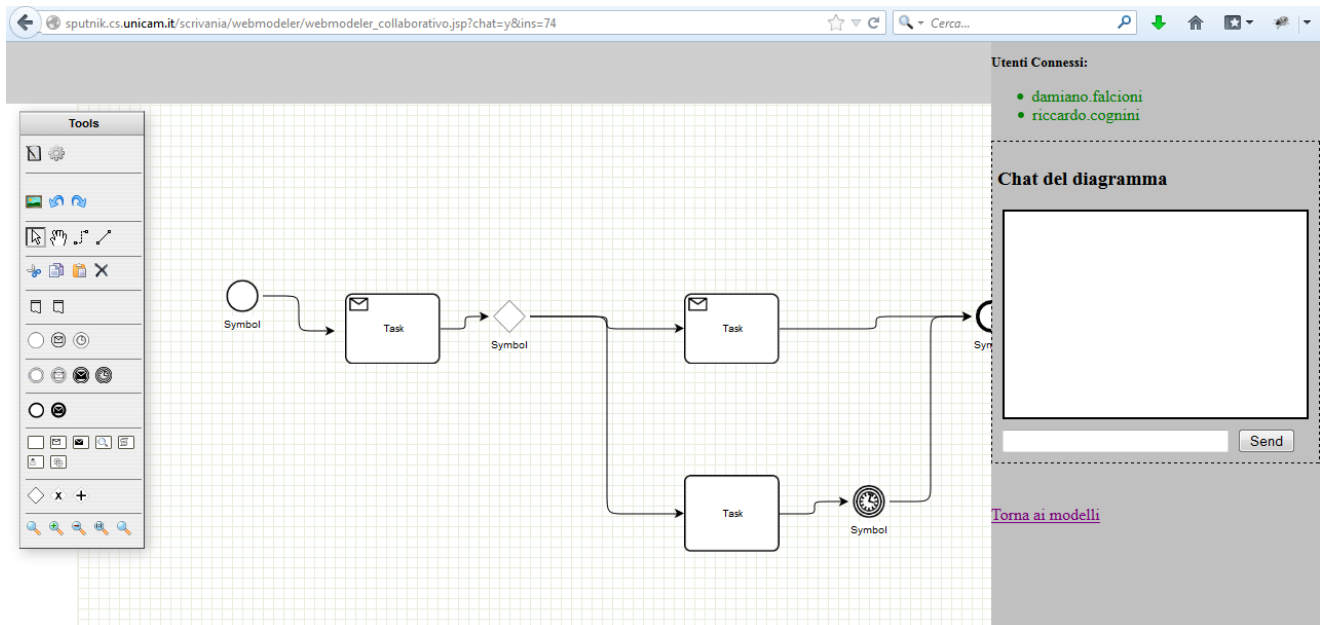
Figure 5. *Scrivania* collaborative BP environment – the BPMN 2.0 modeler

authentication data into smart cards. The security level can be further increase by a personal code that ensures the person from loss and robbery. In particular *Scrivania* supports the Electronic National identity card CNS as the national standard for digital identity card.

- *Administrator Module*, It allows administrators (PA manager) to manage PA services. Two different sub-modules compose this module that are Services Creation sub-module and Manage Services sub-module. The first provides functionalities for services creation; in particular it introduces a collaborative environment for BP modeling in which a BPMN 2.0 model can be designed. It enables communication between editors, during the modeling by a chat system. This sub-module support verification functionalities to guarantee the correctness of the model. Instead, Manage Services sub-module permits administrators to manage services and *Scrivania* functionalities. Using this sub-module, administrators can share services in order to allow citizens to their executions. Administrators can also check statistics about services quality.

- *Citizen Module*, It allows citizens to execute services and to interact each others. Three sub-modules compose this module. (1) Social sub-Module supports citizens communication during services executions. Citizen can also search their friend and give a feedback about a service or single activity of service.(2) Similarity sub-Module promotes services to citizen based on profile similarity. This means that probably two similar citizens need to execute same services. This sub-module provides an algorithm that merges two types of properties: service execution frequency and citizen similarity as a way of validation. (3) Execution sub-Module that provides

functionalities to search and to execute services. It include also the semantic searching functionalities.

In this paper we focalize only on the collaborative functionalities of Services Creation sub-module and on the Execution sub-Module.

### B. Collaboration in the Services Creation Sub-Module

*Scrivania* Services Creation sub-module provide a collaborative BP environment that PA managers use to create PA services in BPMN 2.0 standard. Each service has an owner that is the creator. The owner can enable other PA manager to edit the model during its creation. When the owner enable another PA manger to edit a model, *Scrivania* sends automatically a notification mail to the manager, containing the web link of the shared model. The environment is collaborative because each PA manager can see in real-time how other managers modify a specific PA service.

Figure 5 shows *Scrivania* collaborative BP environment. In the left we can see the toolbar of the modeler, instead in the right there is a list of active users, in this case damiano.falcioni and riccardo.cognini. Below the active user list there is a process chat used by PA manager to communicate.
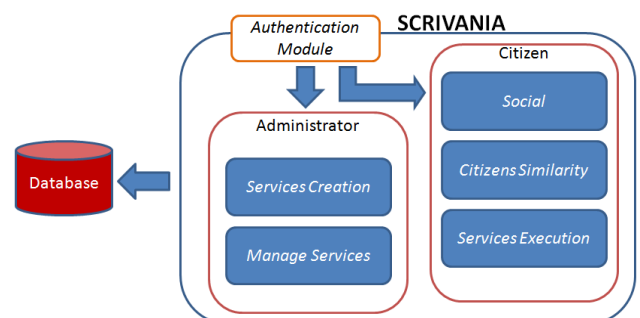


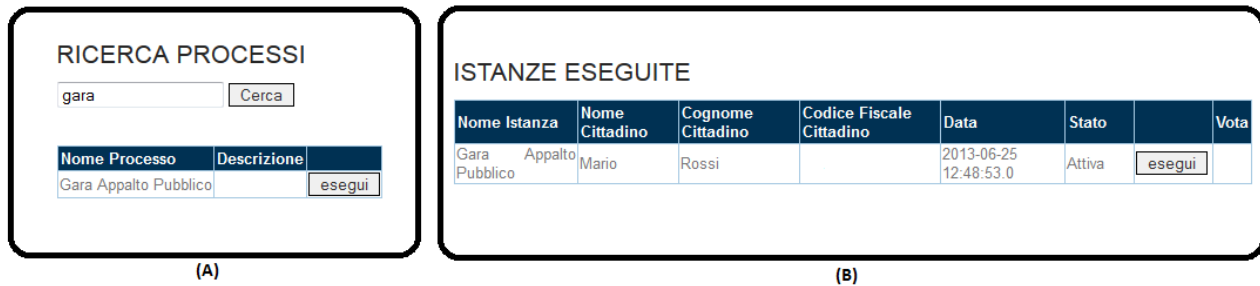Figure 4. *Scrivania* Architecture.

Figure 6. *Scrivania* services research (A) and citizen instance list (B)

### C. Semantic Search of the Execution sub-Module

In order to execute a service, after its creation in the modeling phase, citizens can search it using the semantic search form as Figure 6-A shows. Search results shows the service name and provides a button to allow the process executions. When a citizen clicks on the execution button in his/her home page appear a new service instance as Figure 4-B shows. Each instance has a different instance name (that correspond to services name), a citizen instance, a date and the current status that can be active, in waiting and ended. An instance is active when the citizen still has to execute some tasks, while is in waiting state when he/she has to wait for other participant actions. An ended state means that the service has been completed correctly.

In an Active instance, when a citizen clicks on the execution button (button "esegui" in Figure 6-B) an new page shown in Figure 6-A appears, containing the list of all possible tasks to perform, as Figure 7-A shows. When the citizen choose to execute a task, a new page is opened, where there are some information about the selected task and the semantic search form of the semantic categorization engine as Figure 7-B shows. The search works on a semantic repository. The repository categorizes PA documents (or other kind of document if necessary) over a predefined domain ontology. This ontology has been defined in conjunction with different Italian PA domain expert in order to correctly represent all the information managed in the domain. The steps involved in the categorization of a document on the basis of ontological can be summarized as follows:

1. Text extraction and language identification of the document to be categorized;

2. Division of the text in sentences;

3. Part Of Speech Tagging of each sentence, in order to identify nouns, verbs, adjectives, articles, etc;

4. Dependency analysis to create the syntagms of the sentence and identify subjects, verbs and complements;

5. Comparison between relations defined in the domain ontology and syntagms identified.

6. Memorization of the resulting matches as metadata of the document, in a repository for subsequent search.

The semantic search is then performed as follows:

1. Language identification of the search query;

2. Part Of Speech Tagging of the sentence provided, in order to identify nouns, verbs, adjectives, articles, etc;

3. Dependency analysis to create the syntagms of the sentence and identify subjects, verbs and eventually complements;

4. Comparison between the syntagm identified and the relation stored in the repository and identification of the related documents to return.

5. Delivery of the documents to the user.

The framework used for the semantic features has been build from scratch, merging all the state of the art tools required for a semantic categorization of documents, taking into consideration adaptabilities feature to the Italian language, integration facility and performance. The Part Of Speech tagger and the Dependency Analyzer are in particular trained using a well affirmed open source Italian Corpus Document provided by the research group of the University of Trento.

## VI. CONCLUSION

The paper presents a tool for Public Administration transparency named *Scrivania*. The tool provides to PAs an environment to model they services, in term of Business Process, in collaborative way. Citizen can search and execute the services they need using the execution environment that provides a semantic search in a documents repository in order to provide to citizen a high degree of transparency during the execution of services. Citizens can also use a social environment to communicate and helping each other.

REFERENCES

[1] Bertot, John C. et al (2010) "Using ICTs to create a culture of transparency: E-government and social media as openness and anti-corruption tools for societies". Government Information Quarterly. Vol 27, No 3, pp 264-271.

[2] Lindsay, A. et al, (2003). Business Processes – attempts to find a definition. In Information & Software Technology, Vol. 45, No.15, pp 1015-1019.

[3] Weske, M. (2007). Business process management: concepts, methods, technology. Springer, Berlin, Germany.

[4] Weske, M. (2012). Business process management: concepts, languages, architectures. Springer.

[5] Object Management Group (OMG), 2011. Business Process Model And Notation (BPMN) Version 2.0

[6] Tiwana, A. (2000). The knowledge management toolkit: practical techniques for building a knowledge management system. Prentice Hall PTR.
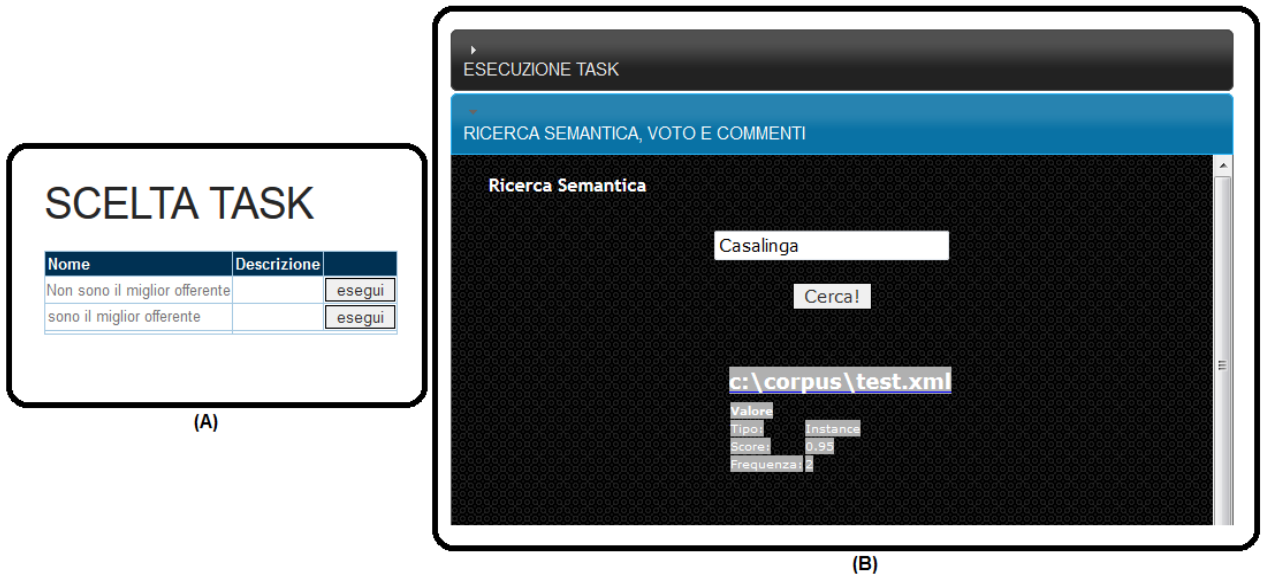
Figure 7. *Scrivania* list of tasks (A) and semantic search form (B)