



TrasparenzAI - Piattaforma per l'analisi e la consultazione della trasparenza amministrativa delle Pubbliche Amministrazioni

Release 1.0.0

Consiglio Nazionale delle Ricerche

12 giu 2025

Contents

1 Panoramica della soluzione	2
2 Architettura della soluzione	3
2.1 Scansione dei siti delle PA	3
3 Componenti principali	5
3.1 Public Sites Service	6
3.2 Config Service	9
3.3 Result Service	10
3.4 Result Aggregator Service	13
3.5 Task Scheduler Service	15
3.6 Rule Service	16
3.7 Conductor service	17
3.8 UI service	21
4 Installazione e configurazione	22
4.1 Autenticazione	22
4.2 Autorizzazione	23
4.3 Config Service	23
4.4 Public Sites Service	24
4.5 Result Service	26
4.6 Result Aggregator Service	26
4.7 Task Scheduler Service	27
4.8 Risorse hardware consigliate	28
5 Manuale Amministratore	30
5.1 Homepage	30
5.2 Menù «Cerca Amministrazioni»	33
5.3 Menù «Mappa delle Amministrazioni»	33
5.4 Funzione «Controlli» - Cronologia controlli e Scan-On-Demand	38
5.5 Funzione «Storico» - Cronologia controlli	38
5.6 Menù «Regole»	44
5.7 Menù «Grafico per regola»	44
5.8 Menù «Mappa per regola»	47
5.9 Menù «Esplora Sezioni»	52
5.10 Menù «Accesso area riservata»	52

6 Appendix	60
6.1 Autori	60
6.2 Glossario	60
Indice	62

Piattaforma opensource per l'analisi e la consultazione della trasparenza amministrativa delle Pubbliche Amministrazioni

La trasparenza amministrativa rappresenta un pilastro fondamentale per il buon funzionamento delle istituzioni pubbliche, garantendo ai cittadini, agli operatori economici e agli organismi di controllo la possibilità di accedere in modo semplice e immediato alle informazioni riguardanti l'attività delle Pubbliche Amministrazioni.

Il Decreto Legislativo 33/2013 ha istituito l'obbligo per le amministrazioni di pubblicare una serie di dati, documenti e informazioni nella sezione *Amministrazione Trasparente*, con l'obiettivo di prevenire fenomeni di corruzione e di favorire una maggiore accountability.

Tuttavia, l'applicazione di questa normativa si è rivelata nel tempo complessa e disomogenea, generando difficoltà sia per le amministrazioni stesse, che devono gestire la pubblicazione e l'aggiornamento dei dati, sia per gli utenti finali, che incontrano ostacoli nell'individuazione e nella consultazione delle informazioni.

A fronte di queste criticità, la **piattaforma TrasparenzAI** (realizzata nell'ambito della collaborazione istituzionale tra *CNR* e *ANAC*) nasce con lo scopo di semplificare e rendere più efficiente l'accesso e il monitoraggio delle informazioni pubblicate nella sezione *Amministrazione Trasparente* dei siti delle amministrazioni pubbliche.

La piattaforma è disponibile all'indirizzo:

- [https://www.traspenzai.it](https://www.trasparenzai.it)

CHAPTER 1

Panoramica della soluzione

La **piattaforma TrasparenzAI** nasce con lo scopo di semplificare e rendere più efficiente l'accesso e il monitoraggio delle informazioni pubblicate nella sezione *Amministrazione Trasparente* dei siti delle amministrazioni pubbliche.

Il problema principale riscontrato è la grande eterogeneità delle modalità di pubblicazione. Ogni amministrazione, pur rispettando formalmente gli obblighi normativi, adotta soluzioni tecniche differenti per l'organizzazione e la struttura della sezione, con variazioni nella denominazione delle sotto-sezioni, nei percorsi di accesso e nella tipologia dei file pubblicati.

Questa frammentazione complica la consultazione delle informazioni e rende estremamente difficoltosa la verifica della loro presenza e conformità. Se per un cittadino alla ricerca di un determinato documento può risultare frustrante dover navigare in siti web strutturati in modo diverso, per *ANAC* diventa ancora più complesso ed oneroso monitorare il rispetto degli obblighi di pubblicazione su scala nazionale.

La piattaforma risponde a queste necessità introducendo un sistema automatizzato in grado di verificare in modo sistematico la presenza e la struttura delle sezioni *Amministrazione Trasparente*, fornendo un quadro aggiornato della situazione a livello nazionale.

Questo strumento non si limita a rilevare le irregolarità, ma offre anche un supporto alle amministrazioni per adeguarsi agli standard richiesti, migliorando così il livello complessivo della trasparenza amministrativa.

L'assenza di un meccanismo centralizzato che permetta di uniformare la pubblicazione delle informazioni, rappresenta un ostacolo alla piena efficacia della normativa vigente e la piattaforma si pone l'obiettivo di colmare questa lacuna attraverso l'adozione di strumenti tecnologici avanzati.

Basata su un'architettura modulare con microservizi dockerizzati, TrasparenzAI è riutilizzabile per progetti basati su tecniche di web scraping programmabile. I componenti software sviluppati sono rilasciati alla comunità con licenza opensource. Con questo progetto, *CNR* e *ANAC* mirano a costruire un sistema di controllo e vigilanza che sia efficiente, flessibile e orientato all'evoluzione digitale delle pubbliche amministrazioni, ponendo le basi per la futura stagione della trasparenza.

Qui puoi trovare tutta la documentazione a corredo del software opensource concesso in riuso (architettura, componenti, installazione econfigurazione, manuale amministratore interfaccia web).

Se sei interessato al codice sorgente della piattaforma lo puoi trovare su github:

- <https://github.com/trasparenzai>

CHAPTER 2

Architettura della soluzione

La piattaforma è composta da molti componenti, integrati in una architettura a microservizi, comunicanti tramite il paradigma REST. Ogni microservizio è realizzato con framework opensource e rilasciato a sua volta come software opensource. Anche tutti i componenti infrastrutturali utilizzati nella piattaforma sono opensource, azzerando i costi di licenza e permettendo una totale riusabilità di questa piattaforma per scopi uguali o analoghi a quelli per cui è stata pensata.

È previsto un accesso differenziato alle funzionalità della piattaforma, tramite un sistema di autenticazione e autorizzazione basato sul protocollo OAuth2 e dei ruoli predefiniti inseriti nel token JWT.

Sia l'interfaccia WEB, realizzata dal componente *UI Service*, che ogni API REST, è integrata quindi con un IDP OAuth2. Il servizio in staging realizzato per ANAC utilizza [Keycloak](#) come IDP OAuth2.

Il grafico in Fig. 2.1 sottostante riassume i componenti principali del sistema.

Rispetto a una tipica architettura a microservizi è stato scelto di non introdurre al momento un API Gateway e un Service Discovery, in quanto la sua adozione potrebbe dipendere dalle politiche di deploy della soluzione. In particolare qualora si decida di usare Kubernetes potrebbe essere indicato utilizzare i meccanismi di API Gateway e Service Discovery disponibili nel coordinatore dei container.

2.1 Scansione dei siti delle PA

La fase di scansione dei siti delle PA viene coordinata dal servizio *Conductor Service*. Questo servizio è basato sul componente opensource [Conductor](#) realizzato da Netflix.

La definizione dei workflow e dei task necessari per compiere tutte le operazioni di analisi, verifica e salvataggio dei risultati delle scansioni dei siti delle PA è definita tramite alcuni file JSON. Questo garantisce una facile configurabilità e adattabilità di questa soluzione a evoluzioni di questa piattaforma oppure a problematiche di crawling e analisi di natura diversa da quella di questo progetto.

L'avvio della fase di scansione di tutti i siti delle PA viene avviata dal microservizio *Task Scheduler Service*, il quale invoca via REST il *Conductor Service* con una cadenza configurabile (per esempio 3 o 4 volte la settimana).

La lista dei siti delle PA è prelevata dal microservizio *Public Site Service*, mentre i risultati delle validazioni sono inseriti sia nel microservizio *Result Service* che nel servizio *Result Aggregator Service*. Inoltre le screenshot delle



Figure2.1: TrasprenzAI - Overview Diagram

pagine HTML ritenute problematiche dal sistema vengono archiviate in uno storage *S3 like*, in particolare nel servizio fornito in staging ad ANAC viene utilizzato il prodotto Opensource **Minio**.

Le regole applicate per la verifica della corrispondenza del sito con la legge sulla trasparenza sono definite tramite il servizio **Rule Service**. Quest'ultimo è stato realizzato in modo da essere un servizio generico di applicazioni di regole di parsing e configurabile tramite file JSON.

Per l'attività di crawling e di rendering delle pagine HTML da analizzare, la soluzione prevede l'utilizzo di un proprio crawler per prelevare lo streaming html delle pagine da analizzare e l'adozione di un **Selenium Hub** per distribuire su più istanze di Google Chrome il rendering delle pagine HTML che contengono codice javascript da interpretare.

CHAPTER 3

Componenti principali

Il sistema TrasparenzAI è di tipo modulare ed è composta sia da componenti sviluppati ad-hoc per il progetto che da software opensource già disponibili.

Di seguito la lista dei componenti sviluppati per il progetto:

- <https://github.com/trasparenzai/public-sites-service>
- <https://github.com/trasparenzai/rule-service>
- <https://github.com/trasparenzai/result-service>
- <https://github.com/trasparenzai/result-aggregator-service>
- <https://github.com/trasparenzai/task-scheduler-service>
- <https://github.com/trasparenzai/conductor>
- <https://github.com/trasparenzai/workflow-definition>
- <https://github.com/trasparenzai/ui-service>

Ogni componente è stato realizzato nell'ottica di un possibile riuso da parte della comunità opensource italiana (per esempio, progetti di crawling e analisi di siti di web simili). I componenti sono stati sviluppati principalmente in Java (con [Spring Boot](#)) e Python ([FastAPI](#) e [Uvicorn](#)) per la parte backend e typescript ([Angular](#)) per la parte frontend.

I software opensource utilizzati sono:

- Keycloak
- Postgresql
- Minio
- Selenium Grid
- Selenium Node Chrome
- Conductor
- Traefik

Del software *Conductor* è stato effettuato un fork per introdurre l'autenticazione come client OAuth2 nei task che interagiscono con le API REST della piattaforma.

3.1 Public Sites Service

Public Sites Service è il componente che si occupa di gestire le informazioni principali relative agli enti pubblici italiani ed in particolare i siti istituzionali.

Public Sites Service mantiene nel proprio datastore locale le informazioni degli enti che possono essere inserite/aggiorinate tramite gli OpenData di *IndicePA*, oppure inserite tramite appositi servizi endpoint REST.

L'idea è quella di avere una fonte facile da consultare e estendibile delle informazioni delle organizzazioni pubbliche da analizzare. In particolare sono trattate automaticamente le info utili derivanti *IndicePA* ma è possibile inserire altri enti da sottoporre a analisi, inserendoli via REST in questo servizio oppure integrando altre fonti esterne sincronizzate automaticamente.

Public Sites Service fornisce alcuni servizi REST utilizzabili in produzione per:

- mostrare la lista degli enti presenti negli OpenData di *IndicePA*
- inserire ed aggiornare all'interno del servizio le informazioni degli Enti tramite gli OpenData di *IndicePA*
- geolocalizzare gli Enti italiani tramite il servizio Nominatim di OpenStreetMap
- visualizzare i dati di un Ente
- mostrare la lista paginata degli Enti presenti nel servizio, con possibilità di filtrarli per codiceCategoria, codiceFiscaleEnte, codiceIpa, denominazioneEnte
- inserire, aggiornare e cancellare le informazioni degli Enti all'interno del servizio (direttamente senza passare da *IndicePA*)

Il servizio sincronizza e rende disponibili via REST anche le informazioni dei comuni italiani, prelevando ogni notte il CSV dal sito dell'ISTAT dei comuni e aggiornando questo info dentro il servizio stesso. Le info dei comuni servono anche per effettuare una geolocalizzazione più precisa degli enti, che su *IndicePA* sono classificati solamente tramite il codice catastale del comune.

L'aggiornamento dei dati locali al servizio Public Sites Service tramite *IndicePA* avviene ogni mattina alle 6:30. L'aggiornamento dei dati locali al servizio Public Sites Service tramite il CSV di ISTAT avviene ogni mattina alle 6:40.

Il codice sorgente di questo componente è disponibile su GitHub:

- <https://github.com/trasparenzai/public-sites-service>

3.1.1 OpenAPI e Swagger UI

Una volta avviato il servizio, i servizi REST sono documentati tramite OpenAPI e consultabili all'indirizzo /swagger-ui/index.html.

3.1.2 Mappa delle PA Italiane

Il servizio contiene anche una mappa geografica delle PA italiane realizzata tramite leaflet.

3.1.3 Sicurezza

Gli endpoint REST di questo servizio sono protetti tramite autenticazione OAuth con Bearer Token. È necessario configurare l'idp da utilizzare per validare i token OAuth tramite le due proprietà mostrate nell'esempio seguente:

The screenshot shows the TrasparenzAI API documentation interface. At the top, there's a navigation bar with the Swagger logo, the URL '/public-sites-service/v3/api-docs', and a green 'Explore' button. Below the header, the title 'Public sites Service' is displayed with a version '0.1.0' and a 'GAS 3.0' badge. A sub-header '(public-sites-service/v3/api-docs)' follows. A brief description states: 'Public Sites Service si occupa di gestire le informazioni principali relative agli enti pubblici italiani ed in particolare i siti istituzionali'. On the left, there's a 'Servers' dropdown set to 'ipublic-sites-service - Public Sites Service URL' and an 'Authorize' button with a lock icon. The main content area is divided into sections: 'Geo Controller' (Visualizzazione e gestione delle informazioni geografiche), 'Admin Controller' (Metodi di supporto per visualizzazione e gestione dei comuni italiani), and 'Company Controller' (Gestione delle informazioni degli Enti). Each section lists various API endpoints with their methods (e.g., POST, GET), URLs, and descriptions. Some endpoints are highlighted with different colors (green, orange, red) and some have a lock icon indicating they require authentication.

Geo Controller

Visualizzazione e gestione delle informazioni geografiche

Mappa Leaflet Pubbliche Amministrazioni Italiane. ^

POST	/v1/geo/updateExistingCompanyAddresses	Aggiornamento della geolocalizzazione degli enti già geolocalizzati con indirizzo di Nominativi di OSM.		
POST	/v1/geo/updateCompanyAddresses	Aggiornamento della geolocalizzazione degli enti senza indirizzo presenti nel servizio tramite Nominativi di OSM.		
POST	/v1/geo/updateCompanyAddressesUsingGoogleMaps	Aggiornamento della geolocalizzazione degli enti presenti nel servizio tramite Google Maps.		
POST	/v1/geo/updateCompanyAddressUsingGoogleMaps	Aggiornamento della geolocalizzazione dell'ente indicato tramite codiceIpa utilizzando il servizio Google Maps.		
POST	/v1/geo/updateCompanyAddress/{id}	Aggiornamento della geolocalizzazione del indirizzo dell'ente trovate tramite OpenstreetMap.		
POST	/v1/geo/fixCompaniesWithoutMunicipality	Aggiornamento del riferimento al comune degli enti che ne sono sprovvisti.		
GET	/v1/geo/geojson	Indirizzi geolocalizzati presenti nel sistema in formato GeoJson.		
GET	/v1/geo/geoCompanyGoogleMapsAddresses/{id}			
GET	/v1/geo/geoCompanyAddresses/{id}	Visualizzazione di tutte le geolocalizzazioni del indirizzo dell'ente, trovate tramite OpenstreetMap.		
GET	/v1/geo/geoCompanyAddress/{id}	visualizzazione della geolocalizzazione con la maggiore 'Importance' tra le possibili geolocalizzazioni del indirizzo dell'ente trovate tramite OpenstreetMap.		

Admin Controller

Metodi di supporto per visualizzazione e gestione dei comuni italiani

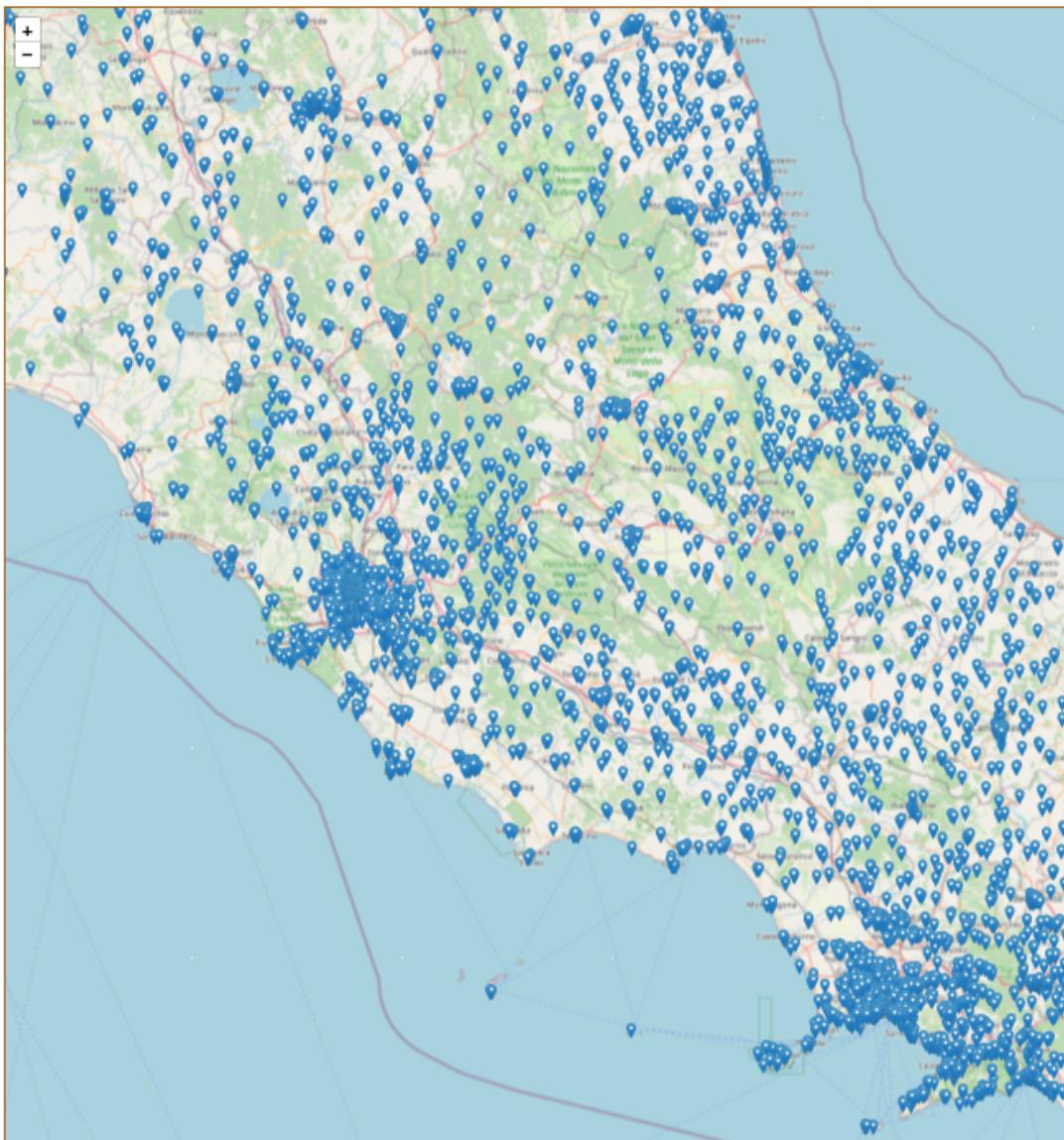
POST	/v1/admin/updateIstatMunicipalities	Aggiornamento degli enti presenti nel servizio tramite IndicePA.		
POST	/v1/admin/updateIndicePaCompanies	Aggiornamento degli enti presenti nel servizio tramite IndicePA.		
GET	/v1/municipalities	Visualizzazione di tutti i comuni attivi presenti nel sistema.		
GET	/v1/municipalities/{id}	Visualizzazione delle informazioni di un comune.		
GET	/v1/admin/istatMunicipalities	Visualizzazione di tutti i comuni presenti nel file CSV di Istat.		
GET	/v1/admin/indicePaCompanies	Visualizzazione di tutti gli enti presenti in IndicePA.		
DELETE	/v1/admin/evictCaches	Svuota le cache utilizzate, in particolare quella degli indirizzi geolocalizzati degli enti.		

Company Controller

Gestione delle informazioni degli Enti

GET	/v1/companies	Visualizzazione di tutti gli enti presenti nel sistema.		
PUT	/v1/companies	Creazione di un ente.		
POST	/v1/companies	Aggiornamento dei dati di un ente.		
GET	/v1/companies/{id}	Visualizzazione delle informazioni di un ente.		
DELETE	/v1/companies/{id}	Eliminazione di un ente.		
GET	/v1/companies/{id}/address	Visualizzazione della geolocalizzazione dell'indirizzo di un ente.		

Mappa Pubbliche Amministrazioni Italiane



```
- spring.security.oauth2.resource-server.jwt.issuer-uri=https://dica33.ba.cnr.it/keycloak/
  ↳ realms/trasparenzai
- spring.security.oauth2.resource-server.jwt.jwk-set-uri=https://dica33.ba.cnr.it/
  ↳ keycloak/realms/trasparenzai/protocol/openid-connect/certs
```

Per l'accesso in HTTP GET all'API è sufficiente essere autenticati, per gli endpoint accessibili con PUT/POST/DELETE è necessario oltre che essere autenticati che il token OAuth contenga un role ADMIN o SUPERUSER.

3.2 Config Service

Config Service è il componente che si occupa di archiviare e distribuire alcune informazioni di configurazione dei servizi che compongono lo stack del progetto TrasparenzAI.

Config Service mantiene nel proprio datastore locale le configurazioni che sono fornite agli altri microservizi. Le configurazioni possono essere inserite/aggiorinate tramite gli appositi endpoint REST presenti in questo servizio.

Le configurazioni disponibili sono fornite sia sotto forma di endpoint REST con le relative CRUD, che nel formato utilizzato da *Spring Cloud Config* attraverso il path **/config**, inserendo il nome del servizio e il profilo richiesto nell'url, come per esempio:

```
$ http GET :8888/config/task-scheduler/default

{
  "label": null,
  "name": "task-scheduler",
  "profiles": [
    "default"
  ],
  "propertySources": [
    {
      "name": "task-scheduler-default",
      "source": {
        "tasks.fake.cron.expression": "0 46 15 * * ?",
        "test.property1": "testme"
      }
    }
  ],
  "state": null,
  "version": null
}
```

I microservizi Spring che vogliono utilizzare questo servizio di configurazione centralizzato possono farlo specificando nella propria configurazione tre parametri tipo:

```
spring.config.import=optional:configserver:http://localhost:8888/config
spring.cloud.config.username=config-service-user
spring.cloud.config.password=PASSWORD_DA_IMPOSTARE_E_CONDIVIDERE_CON_I_CLIENTI
```

Dove naturalmente va impostato il corretto URL a cui risponde questo servizio.

Il codice sorgente di questo componente è disponibile su GitHub:

- <https://github.com/trasparenzai/config-service>

3.2.1 OpenAPI e Swagger UI

Una volta avviato il servizio i servizi REST sono documentati tramite OpenAPI e consultabili all'indirizzo /swagger-ui/index.html.

3.2.2 Sicurezza

L'accesso in lettura alla configurazione di tipo *Spring Cloud Config* disponibile al path **/config** è protetto con autenticazione di tipo *Basic Auth*, i microservizi che vogliono utilizzare questo path per ottenere la configurazione devono utilizzare l'utente e la password specificati tramite i parametri *spring.security.user.name* e *spring.security.user.password*, i quali possono essere specificati nel *docker-compose.yml* come nell'esempio seguente:

- `spring.security.user.name=config-service-user`
- `spring.security.user.password=PASSWORD_DA_IMPOSTARE_E_CONDIVIDERE_CON_I_CLIENT`

Invece gli endpoint REST di questo servizio disponibili al path **/properties** sono protetti tramite autenticazione OAuth con Bearer Token. È necessario configurare l'IDP da utilizzare per validare i token OAuth tramite le due proprietà impostabili nel *docker-compose.yml* come nell'esempio seguente:

- `spring.security.oauth2.resourceserver.jwt.issuer-uri=https://dica33.ba.cnr.it/keycloak/realm/trasparenzai`
- `spring.security.oauth2.resourceserver.jwt.jwk-set-uri=https://dica33.ba.cnr.it/keycloak/realm/trasparenzai/protocol/openid-connect/certs`

Per l'accesso in HTTP GET all'API è sufficiente essere autenticati, per gli endpoint accessibili con PUT/POST/DELETE è necessario oltre che essere autenticati che il token OAuth contenga un role ADMIN o SUPERUSER.

3.3 Result Service

Result Service è il componente che si occupa di gestire i risultati delle verifiche sulla corrispondenza dei siti degli enti pubblici italiani in relazione al decreto legge 33/2013 sulla transparenza.

Result Service fornisce alcuni servizi REST utilizzabili in produzione per:

- inserire, aggiornare e cancellare all'interno del servizio le informazioni di una verifica effettuata su un sito web di una PA
- visualizzare i dati di una verifica su un sito web
- mostrare la lista delle verifiche effettuate
- esportare in CSV i risultati delle validazioni presenti

Il codice sorgente di questo componente è disponibile su GitHub:

- <https://github.com/trasparenzai/result-service>

3.3.1 OpenAPI e Swagger UI

Una volta avviato il servizio i servizi REST sono documentati tramite OpenAPI e consultabili all'indirizzo /swagger-ui/index.html.

3.3.2 Sicurezza

Gli endpoint REST di questo servizio sono protetti tramite autenticazione OAuth con Bearer Token. E' necessario configurare l'idp da utilizzare per validare i token OAuth tramite le due proprietà mostrare nell'esempio seguente:

The screenshot shows the Swagger UI interface for the config-service API. At the top, it displays the title "OpenAPI definition" with a "v0" version indicator and a "OAS 3.0" badge. Below the title, there is a URL field containing "http://edica33.ha.cnr.it/config-service". The main content area is organized into sections for different controllers:

- profile-controller**:
 - GET /profile
 - GET /profile/properties
- property-entity-controller**:
 - GET /properties
 - POST /properties
 - GET /properties/{id}
 - PUT /properties/{id}
 - DELETE /properties/{id}
 - PATCH /properties/{id}
- encryption-controller**:
 - POST /config/encrypt
 - POST /config/encrypt/{name}/{profiles}
 - POST /config/decrypt
 - POST /config/decrypt/{name}/{profiles}
 - GET /config/key
 - GET /config/key/{name}/{profiles}
 - GET /config/encrypt/status
- environment-controller**:
 - GET /config/{name}/{profiles}/{label}
 - GET /config/{name}/{profiles}
 - GET /config/{name}-{profiles}.yaml
 - GET /config/{name}-{profiles}.yml
 - GET /config/{name}-{profiles}.properties
 - GET /config/{name}-{profiles}.json
 - GET /config/{label}/{name}-{profiles}.yaml
 - GET /config/{label}/{name}-{profiles}.yml
 - GET /config/{label}/{name}-{profiles}.properties
 - GET /config/{label}/{name}-{profiles}.json

The screenshot shows the Swagger UI interface for the Transparency Results Service. At the top, there's a header with the Swagger logo, the URL '/result-service/v3/api-docs', and a green 'Explore' button. Below the header, the title 'Transparency Results Service' is displayed with a version of '0.2.0' and an 'OAS 3.0' badge. A sub-header notes that the service handles validation results for Italian public websites according to decree 33/2013. The main content area is titled 'Result Controller' and describes it as 'Gestione delle informazioni dei risultati di validazione dei siti delle PA'. It lists various API endpoints with their methods, URLs, descriptions, and status indicators (e.g., lock icons). The endpoints include:

- GET /v1/results: Visualizzazione dei risultati di validazione presenti nel sistema, filtrabili utilizzando alcuni parametri.
- PUT /v1/results: Creazione di un risultato di validazione.
- POST /v1/results: Aggiornamento dei dati di un risultato di validazione.
- PUT /v1/results/bulk: Creazione di più di un risultato di validazione.
- GET /v1/results/{id}: Visualizzazione delle informazioni di un risultato di validazione.
- DELETE /v1/results/{id}: Eliminazione di un risultato di validazione.
- GET /v1/results/lastRunAsCsv: visualizzazione dei risultati dell'ultima validazione registrata nel sistema.
- GET /v1/results/lastResult: Visualizzazione delle informazioni del ultimo risultato memorizzato nel sistema.
- GET /v1/results/csv: Visualizzazione dei risultati di validazione presenti nel sistema.
- GET /v1/results/countAndGroupByWorkflowIdAndStatus: visualizzazione delle informazioni presenti nel sistema raggruppate per flusso e stato della regola applicata.
- GET /v1/results/all: Visualizzazione dei risultati di validazione presenti nel sistema, filtrabili utilizzando alcuni parametri.
- DELETE /v1/results/byWorkflow/{id}: Eliminazione dei risultati di validazione associati a un workflow id.

```
- spring.security.oauth2.resource-server.jwt.issuer-uri=https://dica33.ba.cnr.it/keycloak/
  ↵realms/trasparenzai
- spring.security.oauth2.resource-server.jwt.jwk-set-uri=https://dica33.ba.cnr.it/
  ↵keycloak/realms/trasparenzai/protocol/openid-connect/certs
```

Per l'accesso in HTTP GET all'API è sufficiente essere autenticati, per gli endpoint accessibili con PUT/POST/DELETE è necessario oltre che essere autenticati che il token OAuth contenga un role ADMIN o SUPERUSER.

3.4 Result Aggregator Service

Result Aggregator Service è il componente che si occupa di gestire i risultati delle verifiche sulla corrispondenza, aggregando i risultati di validazione con altre informazioni sugli enti pubblici prelevate da altri servizi.

Result Aggregator Service fornisce alcuni servizi REST utilizzabili in produzione per:

- inserire, aggiornare e cancellare all'interno del servizio le informazioni di una verifica effettuata su un sito web di una PA ed dei dati geografici degli enti pubblici
- esportare in geoJson i risultati delle validazioni presenti arricchiti con la geolocalizzazione degli enti

Il codice sorgente di questo componente è disponibile su GitHub:

- <https://github.com/trasparenzai/result-aggregator-service>

3.4.1 OpenAPI e Swagger UI

Una volta avviato il servizio, i servizi REST sono documentati tramite OpenAPI e consultabili all'indirizzo /swagger-ui/index.html.

3.4.2 Dipendenze e configurazione

Questo servizio ha due dipendenze dagli altri componenti per funzionare:

- il **Result Service** da cui leggere le info sulle verifiche
- il **Public Site Service** da cui prelevare le info geografiche delle PA

L'indirizzo di entrambi questi servizi è configurabile nel file `application.properties` oppure tramite variabili d'ambiente se avviato tramite Docker.

3.4.3 Sicurezza

Gli endpoint REST di questo servizio sono protetti tramite autenticazione OAuth con Bearer Token. È necessario configurare l'idp da utilizzare per validare i token OAuth tramite le due proprietà mostrare nell'esempio seguente:

```
- spring.security.oauth2.resource-server.jwt.issuer-uri=https://dica33.ba.cnr.it/keycloak/
  ↵realms/trasparenzai
- spring.security.oauth2.resource-server.jwt.jwk-set-uri=https://dica33.ba.cnr.it/
  ↵keycloak/realms/trasparenzai/protocol/openid-connect/certs
```

Per l'accesso in HTTP GET all'API è sufficiente essere autenticati, per gli endpoint accessibili con PUT/POST/DELETE è necessario oltre che essere autenticati che il token OAuth contenga un role ADMIN o SUPERUSER.

Inoltre questo servizio interagisce con il `_result_service_` e il `_public_site_service_` per prelevare i risultati da aggregare.

The screenshot shows the Swagger UI interface for the Transparency Results Service API. At the top, there's a header with the Swagger logo, the URL '/result-service/v3/api-docs', and a green 'Explore' button. Below the header, the title 'Transparency Results Service' is displayed with a version of '0.2.0' and an 'OAS 3.0' badge. A descriptive text states: 'Transparency Results Service si occupa di gestire i risultati delle verifiche di conformità sulla legge della trasparenza del decreto legge 33/2013 per i siti degli enti pubblici italiani.' Underneath, there's a 'Servers' section with a dropdown menu set to 'iresult-service - Transparency Results Service URL' and an 'Authorize' button. The main content area is titled 'Result Controller' and describes it as 'Gestione delle informazioni dei risultati di validazione dei siti delle PA'. It lists various API endpoints with their methods, URLs, and descriptions:

- GET /v1/results** Visualizzazione dei risultati di validazione presenti nel sistema, filtrabili utilizzando alcuni parametri.
- PUT /v1/results** Creazione di un risultato di validazione.
- POST /v1/results** Aggiornamento dei dati di un risultato di validazione.
- PUT /v1/results/bulk** Creazione di più di un risultato di validazione.
- GET /v1/results/{id}** Visualizzazione delle informazioni di un risultato di validazione.
- DELETE /v1/results/{id}** Eliminazione di un risultato di validazione.
- GET /v1/results/lastRunAsCsv** visualizzazione dei risultati dell'ultima validazione registrata nel sistema.
- GET /v1/results/lastResult** Visualizzazione delle informazioni del ultimo risultato memorizzato nel sistema.
- GET /v1/results/csv** Visualizzazione dei risultati di validazione presenti nel sistema.
- GET /v1/results/countAndGroupByWorkflowIdAndStatus** visualizzazione delle informazioni presenti nel sistema raggruppate per flusso e stato della regola applicata.
- GET /v1/results/all** Visualizzazione dei risultati di validazione presenti nel sistema, filtrabili utilizzando alcuni parametri.
- DELETE /v1/results/byWorkflow/{id}** Eliminazione dei risultati di validazione associati a un workflow id.

Per configurare il client REST che accede a questi due servizi è necessario configurare questi parametri nel *docker-compose.yml*, in particolare verificare *client-id*, *client-secret* e *issuer-uri*.

Esempio di configurazione dell'environment nel docker-compose.yml:

```
# Generare un Service Account Oidc con questo client-id, oppure cambiare questo valore
- spring.security.oauth2.client.registration.oidc.client-id=result-aggregator
# Client Secret da generare nel Identity Provider e impostare qui
- spring.security.oauth2.client.registration.oidc.client-secret=client_secret_da_generare
# URL dell'issuer OIDC da impostare
- spring.security.oauth2.client.provider.oidc.issuer-uri=https://dica33.ba.cnr.it/
  ↳keycloak/realms/trasparenzai
# - spring.security.oauth2.client.registration.oidc.authorization-grant-type=client_
  ↳credentials #DEFAULT
# - spring.security.oauth2.client.registration.oidc.scope=openid #DEFAULT
# - spring.security.oauth2.client.registration.oidc.provider=oidc #DEFAULT
```

3.5 Task Scheduler Service

Task Scheduler Service è il componente che si occupa di avviare alcuni processi eseguiti a intervalli fissi, come per esempio l'**avvio delle scansioni** dei siti del PA per la verifica della corrispondenza dei requisiti e la **cancellazione dei risultati di scansione più vecchi**.

Nell'utilizzo tramite **docker-compose.yml** ricordarsi di impostare nel **.env** la corretta variabile d'ambiente che specifica l'url del config-service da utilizzare e la password per l'autenticazione Basic Auth con il *config-service*:

```
environment:
- confighost=${CONFIG_HOST}
- spring.security.oauth2.client.registration.oidc.client-secret=${OIDC_CLIENT_SECRET}
```

Le informazioni di configurazione dei cron relativi ai workflow possono essere visualizzate all'url **/tasks/workflowCronConfig**.

Il codice sorgente di questo componente è disponibile su **GitHub**:

- <https://github.com/trasparenzai/task-scheduler-service>

3.5.1 OpenAPI e Swagger UI

Una volta avviato il servizio, i servizi REST sono documentati tramite OpenAPI e consultabili all'indirizzo **/swagger-ui/index.html**.

3.5.2 Dipendenze e configurazione

Questo servizio ha quattro dipendenze per funzionare:

- il **Config Service** da cui prelevare i parametri per l'avvio dei nuovi flussi e la configurazione con le policy di cancellazione dei vecchi risultati
- il **Conductor Service** per avviare nuovi flussi di scansioni dei siti, per prelevare la lista dei flussi terminati e per cancellare dal Conductor i flussi più vecchi
- il **Result Service** per cancellare i risultati di validazione più vecchi
- il **Result Aggregator Service** per cancellare i risultati di validazione aggregati più vecchi

L'indirizzo di questi servizi è configurabile nel file **application.properties** oppure tramite variabili d'ambiente se avviato tramite Docker.

The screenshot shows the Task Scheduler Service OpenAPI documentation. At the top, there's a header with the Swagger logo, the URL '/task-scheduler-service/v3/api-docs', and a green 'Explore' button. Below the header, the title 'Task Scheduler Service OpenAPI' is displayed with a version of '0.1.0' and 'OAS 3.0'. A link to '/task-scheduler-service/v3/api-docs' is also present. The main content area is titled 'task-info-controller' and lists several API endpoints:

- POST /tasks/startWorkflow** (green background)
- GET /tasks/workflowIdsToPreserveFromConfig** (blue background)
- GET /tasks/workflowCronConfig** (blue background)
- GET /tasks/expiredWorkflows** (blue background)
- GET /tasks/completedWorkflows** (blue background)
- DELETE /tasks/deleteWorkflow** (red background)
- DELETE /tasks/deleteExpiredWorkflows** (red background)

On the right side of the interface, there are 'Servers' and 'Authorize' buttons.

3.6 Rule Service

Rule Service è il componente che definisce ed implementa le regole relative al D.Lgs. n. 33-2013 sulla trasparenza nella PA.

Fornisce l'albero delle regole definito in `application.yaml` oppure all'interno del `Config Service`, che quindi può essere modificato o ampliato sia attraverso variabili di ambiente o della JVM prima di avviare il servizio, oppure aggiornando la configurazione per poi successivamente invocare l'endpoint dell'`actuator` per recepire le modifiche.

È possibile interagire con il servizio attraverso degli endpoint REST che permettono la consultazione dell'albero delle regole in formato **json**, e la verifica di una o più regole su un contenuto **html**

3.6.1 Docker

Il servizio è dotato del plugin **jib** che permette di effettuare la **build** tramite `gradle` e fornisce l'immagine per l'esecuzione tramite `docker`

```
./gradlew jibDockerBuild  
docker run -p 8080:8080 -ti rule-service:{version}
```

La documentazione relativa ai servizi REST è consultabile [qui](#) ed è possibile interagire con i servizi attraverso **Swagger** alla seguente [URL](#), inoltre è possibile visualizzare l'albero delle regole in formato **json**

Oppure verificare la regola **root** ad esempio attraverso una **cURL** con un **html** di esempio:

```
curl -X POST http://localhost:8080/v1/rules -H 'Content-type:application/json' --data  
→ 'PGh0bWw+CiAgICA8aGVhZD4KICAgICAgICA8dG10bGU+R2VuZXJpY2EgQW1taW5pc3RyYXppb25lPC90aXRsZT4KICAgIDwvaGVh  
→ '| jq .
```

In alternativa scaricare il contenuto del Sito istituzionale di una Pubblica Amministrazione

```
curl "https://www.anticorruzione.it" | base64 > base64.html
curl -X POST http://localhost:8080/v1/rules -H 'Content-type:application/json' --data-binary @base64.html | jq .
```

La risposta **json** del servizio:

```
{  
    "url": "https://www.anticorruzione.it/amministrazione-trasparente",  
    "ruleName": "amministrazione-trasparente",  
    "term": "Amministrazione Trasparente",  
    "content": "Amministrazione Trasparente",  
    "where": "text",  
    "leaf": false,  
    "status": 200,  
    "score": 4.3884144  
}
```

3.7 Conductor service

Il coordinatore è basato su [Netflix - Conductor](#) una piattaforma gratuita e open source per l'orchestrazione dei microservizi, attraverso flussi di lavoro che definiscono le interazioni tra servizi, il progetto principale è stato forkato per permettere e gestire l'autenticazione e l'autorizzazione sull'esecuzione dei flussi e sul passaggio della stessa ai microservizi invocati dal flusso.

Nel progetto [workflow-definition](#) sono presenti le definizioni in formato *json* dei flussi necessari al completamento degli obiettivi del progetto.

3.7.1 Flusso principale - Amministrazione Trasparente

● Importante

Il flusso [principale](#) necessita dei seguenti parametri di input per la sua corretta invocazione come mostrato in [Tabella 3.1](#)

Table3.1: Parametri di Input per il flusso principale

Nome	Descrizione	Valore consigliato/defaut	Può essere Vuoto?
page_size	Dimensione della pagina per il recupero delle PA	2000	No
parent_workflow	Identificativo del flusso, viene valorizzato con UUID generato	vuoto	Si
codice_categoria	Se valorizzato filtra le PA che fanno parte della categoria	vuoto	Si
codice_ipa	Se valorizzato individua la singola PA	vuoto	Si
crawling_mode	Modalità base di esecuzione del crawler può assumere i valori <i>httpStream</i> e <i>htmlSource</i>	httpStream	No
crawler_save_obj	Booleano indica se salvare sempre la pagina HTML	false	No
crawler_save_scr	Booleano indica se salvare sempre lo screenshot della pagina	false	No
rule_name	Nome della regola		amministrazione-trasparente
root_rule	Nome della regola di base dell'albero		amministrazione-trasparente
execute_child	Booleano indica se controllare le regole figlie	true	No
id_ipa_from	Identificativo numerico della PA da cui partire	0	No
connection_timeout	Timeout in millisecondi della connessione	60000	No
read_timeout	Timeout in millisecondi della lettura	60000	No
connection_timeout_m	Timeout in millisecondi della connessione	120000	No
read_timeout_ma	Timeout in millisecondi della lettura	120000	No
crawler_child_type	Modalità di esecuzione dei flussi figli può assumere i valori <i>SUB_WORKFLOW</i> e <i>START_WORKFLOW</i>	START_WORKFLOW	No
rule_base_url	URL di base del microservizio delle Regole	URL	No
public_company_url	URL di base del microservizio delle PA	URL	No
result_aggregato	URL di base del microservizio Aggregato	URL	No
result_base_url	URL di base del microservizio dei Risultati	URL	No
crawler_uri	URL di base del microservizio Crawler	URL	No

3.7.2 Dettagli del flusso principale

Il primo **TASK** del flusso si occupa di invocare l'aggiornamento della configurazione del microservizio delle regole. Dopo aver valorizzato la variabile necessaria al controllo delle pagine elaborate, il flusso invoca il **microservizio delle PA** descritto in *Public Sites Service* e recupera le informazioni necessarie.

Il blocco recuperato contenente le informazioni di *n* PA viene parcellizzato in base al parametro fornito in input **page_size** e diviso per *10*, utilizzando infine il **TASK FORK/JOIN** vengono eseguiti in parallelo *10* istanze del flusso Rule valorizzando il parametro in input **companies**.

All'uscita del **TASK delle PA**, se il flusso è stato eseguito non per una singola PA, allora vengono rielaborati i risultati con i codici *400* e *407* con i timeout massimi ed eseguiti i flussi **Crawler Result Failed**.

Infine viene eseguito il **TASK** per elaborare la Mappa geolocalizzata dei risultati.

In Fig. 3.1 l'immagine del flusso:

3.7.3 Flusso per singola Amministrazione

Il flusso **Rule Detail** come mostrato in Fig. 3.2 viene eseguito per una singola PA passata come parametro in input **ipa**, controlla inizialmente la **presenza della URL istituzionale** e successivamente invoca il **crawler** il cui risultato viene passato al **microservizio delle regole** e la cui risposta è utilizzata come input al **Task dei risultati**



Figure 3.1: Flusso principale - *Amministrazione Trasparente*



Figure3.2: Flusso per singola Amministrazione

3.8 UI service

UI Service integra e mostra i dati presenti nei vari servizi fornendo la possibilità, avendo gli opportuni permessi, di attivare le funzionalità preposte all'inserimento e alla cancellazione degli stessi, inoltre è possibile attivare l'autenticazione su tutte le pagine, o in alternativa di accedere senza autenticazione per la sola consultazione dei dati, per poi richiederla successivamente.

L'interfaccia web è progettata seguendo le linee guida dell'**accessibilità**, della **responsività** e del **design istituzionale**, secondo quanto previsto da [Designers Italia](#). L'aspetto è professionale, ordinato e coerente con la comunicazione della Pubblica Amministrazione.

CHAPTER 4

Installazione e configurazione

Prima di cominciare assicurati di avere a disposizione sufficiente risorse di CPU, memoria RAM e spazio disco per l'utilizzo desiderato della piattaforma. Controlla la sezione [Risorse hardware consigliate](#) per verificare il dimensionamento necessario in produzione.

La modalità consigliata di installazione è tramite **Docker**, assicurati che su ogni server sui cui effettuare il deploy dei componenti dell'architettura sia installato sia Docker, che Docker Compose.

La piattaforma è composta da diversi servizi e microservizi, esposti solitamente tramite interfacce REST via HTTP/HTTPS. L'esposizione dei servizi/microservizi pubblici tramite protocollo cifrato HTTPS è fortemente consigliata, è possibile utilizzare a questo scopo uno dei vari proxy http/https disponibili. Per la piattaforma fornita in staging è stato utilizzato **Traefik**.

4.1 Autenticazione

La piattaforma necessita di un Identity Provider OAuth2 per l'autenticazione e autorizzazione nell'accesso ai componenti dell'architettura.

Nell'ambiente di staging è stato utilizzato Keycloak come Identity Provider ma un qualunque IDP compatibile OAuth2 può andare bene.

Per configurare i vari componenti è necessario procurarsi l'endpoint per ottenere il token jwt e l'endpoint contenente i certificati pubblici del IDP, per esempio:

- `jwt.issuer-uri` -> <https://dica33.ba.cnr.it/keycloak/realm/trasparenza>
- `jwt.jwk-set-uri` -> <https://dica33.ba.cnr.it/keycloak/realm/trasparenza/protocol/openid-connect/certs>

Sarà necessario impostare questi due parametri nei vari microservizi, come spiegato nel seguito.

Ci sono due tipologie di accesso ai servizi della piattaforma, quello degli utenti (le persone fisiche) e quello dei client (i vari componenti si autenticano se devono comunicare tra di loro).

Per quanto riguarda i client è necessario creare tre *Service Account* di tipo *OpenId Connect* e autenticazione di tipo **client_credentials**, i tre client_id devono essere:

- crawler
- result-aggregator
- task-scheduler

I valori dei rispettivi *client secret* dovrà essere impostato nei microservizi *crawler-service*, *result-aggregator-service* e *task-scheduler-service*.

A questi tre service account deve inoltre essere assegnato un *Service Account Role* di tipo **ROLE_SUPERUSER**.

È inoltre necessario creare un client, sempre di tipo *OpenId Connect*, per l'interfaccia Web Angular JS, il client si deve chiamare **angular-public** e deve avere impostato come **valid redirect url** il valore https://www.trasparenzai.it/*.

4.2 Autorizzazione

L'accesso all'interfaccia web è condizionato dalla presenza o meno di determinati ruoli nel token JWT fornito dal sistema di autenticazione. I ruoli attualmente previsti sono:

- ROLE_USER
- ROLE_ADMIN
- ROLE_SUPERUSER

Le funzionalità mostrate nell'interfaccia web cambiano in funzione del ruolo dell'utente, è quindi necessario attribuire nel Identity Provider OAuth2 il ruolo desiderato ai propri utenti.

4.3 Config Service

Config Service è il componente che si occupa di archiviare e distribuire alcune informazioni di configurazione dei servizi che compongono lo stack del progetto TrasparenzAI.

Config Service mantiene nel proprio datastore locale le configurazioni che sono fornite agli altri microservizi.

Il codice sorgente è disponibile su github:

- <https://github.com/trasparenzai/config-service>

Nel repository github è compreso anche un script per la prima installazione del servizio `first-setup.sh`.

In particolare è necessario configurare la sezione della sicurezza.

4.3.1 Sicurezza

L'accesso in lettura alla configurazione di tipo Spring Cloud Config disponibile al path `/config` è protetto con autenticazione di tipo Basic Auth, l'utente e la password possono essere indicati nel `docker-compose.yml` come nell'esempio seguente:

```
- spring.security.user.name=config-service-user  
- spring.security.user.password=PASSWORD_DA_IMPOSTARE_E_CONDIVIDERE_CON_I_CLIENT
```

Invece gli endpoint REST di questo servizio disponibili al path `/properties` sono protetti tramite autenticazione OAuth con Bearer Token. È necessario configurare l'IDP da utilizzare per validare i token OAuth tramite le due proprietà impostabili nel `docker-compose.yml` come nell'esempio seguente:

```
- spring.security.oauth2.resourceserver.jwt.issuer-uri=https://dica33.ba.cnr.it/keycloak/  
- realms/trasparenzai  
- spring.security.oauth2.resourceserver.jwt.jwk-set-uri=https://dica33.ba.cnr.it/  
- keycloak/realms/trasparenzai/protocol/openid-connect/certs
```

I valori dei parametri *jwt.issuer-uri* e *jwk-set-uri* sono quelli già descritti nella sezione [Autenticazione](#).

4.3.2 Configurazione di default

Il config-service viene fornito con una configurazione predefinita da personalizzare secondo le proprie esigenze.

In particolare sono presenti alcune URL degli altri microservizi che è necessario configurare secondo il proprio setup.

I dati di default possono essere modificati sia tramite l'API REST del servizio che tramite l'interfaccia web (il componente UI Service).

Nell'esempio seguente viene mostrata la configurazione predefinita modificabile direttamente tramite l'interfaccia web della piattaforma.

The screenshot shows the 'PARAMETRI' configuration page. At the top, there is a URL field containing <https://monitorai.ba.cnr.it/conductor-server/api/workflow>. Below the URL, there is a section titled 'PARAMETRI' with a blue information icon. It includes fields for 'Dimensione pagina' (set to 2000) and 'CATEGORIA'. There are several configuration options with checkboxes:

- 'Esegui regole figlie' (checked)
- 'Salva pagina sempre' (checked)
- 'Salva screenshot sempre' (unchecked)

Below these are timeout settings:

- 'Timeout di connessione iniziale...' set to 30000
- 'Timeout di risposta iniziale...' set to 60000
- 'Timeout di connessione massima...' set to 60000
- 'Timeout di risposta massima...' set to 120000

Under 'Modalità di esecuzione dei flussi figli', there is a dropdown menu set to 'START WORKFLOW'. Below it, there are two sections with URLs:

- 'URL del servizio Risultati': <https://dica33.ba.cnr.it/result-service>
- 'URL del servizio Crawler': <http://150.145.95.77:8080/crawl>
- 'URL del servizio delle Regole': <https://monitorai.ba.cnr.it/rule-service>
- 'URL del servizio dei dati Aggregati': <https://dica33.ba.cnr.it/result-aggregator-service>

At the bottom, there is a slider for 'Numero di controlli da conservare' set to 9, and a section for 'Controlli da conservare' with two entries:

- 'Controllo del 28/12/2024 21:00:17, stato COMPLETATO'
- 'Controllo del 05/03/2025 19:00:00, stato COMPLETATO'

In particolare sono sicuramente da impostare:

- URL del servizio Risultati
- URL del servizio Crawler
- URL del servizio delle Regole
- URL del registro delle PA
- URL del servizio dei dati Aggregati

Nel caso si voglia modificare le tempistiche e la frequenza delle scansioni complete di tutte le PA è possibile utilizzare sempre l'interfaccia web.

4.4 Public Sites Service

Public Sites Service è il componente che si occupa di gestire le informazioni principali relative agli enti pubblici italiani ed in particolare i siti istituzionali.

Public Sites Service mantiene nel proprio datastore locale le informazioni degli enti che possono essere inserite/aggiornate tramite gli OpenData di *IndicePA*, oppure inserite tramite appositi servizi endpoint REST. Il servizio utilizza Nominatim di OpenStreetMap per la geolocalizzazione degli indirizzi degli enti pubblici.

Il codice sorgente è disponibile su [github](#):

Espressione per l'esecuzione di un nuovo controllo, la prossima esecuzione avverrà il 19/03/2025 alle ore 19, la successiva il 22/03/2025 alle ore 19

Ore Giorni Mesi

Conferma

Ogni ora

Ogni ora/e a partire dalle ore

Ora specifica (sceglierne una)

0 1 2 3 4 5 6 7 8 9 10 11
 12 13 14 15 16 17 18 19 20 21 22 23

Ogni ora tra le ore e ora

- <https://github.com/trasparenzai/public-sites-service>

Nel repository github è compreso anche uno script per la prima installazione del servizio `first-setup.sh`.

In particolare è necessario configurare la sezione della sicurezza.

4.4.1 Sicurezza

Gli endpoint REST di questo servizio sono protetti tramite autenticazione OAuth con Bearer Token. È necessario configurare l'IDP da utilizzare per validare i token OAuth tramite le due proprietà impostabili nel `docker-compose.yml` come nell'esempio seguente:

```
- spring.security.oauth2.resourceserver.jwt.issuer-uri=https://dica33.ba.cnr.it/keycloak/  
- realms/trasparenzai  
- spring.security.oauth2.resourceserver.jwt.jwk-set-uri=https://dica33.ba.cnr.it/  
- keycloak/realms/trasparenzai/protocol/openid-connect/certs
```

I valori dei parametri `jwt.issuer-uri` e `jwk-set-uri` sono quelli già descritti nella sezione [Autenticazione](#).

4.4.2 Integrazione API Google Maps

Il servizio è già predisposto per l'integrazione con l'API di Google Maps per la geocalizzazione degli indirizzi degli enti pubblici. L'API Google Maps fornisce solitamente una migliore individuazione delle coordinate GPS degli indirizzi indicati nel [IndicePA](#). L'API Google Maps è però a pagamento, con un freetier per un numero iniziale di ricerche, è necessario procurarsi una Google Maps Key per poter utilizzare questo servizio, la quale richiede di inserire una carta di credito per gli eventuali pagamenti oltre il freetier.

L'utilizzo della API Google Maps può essere attivata nel public sites service impostando questo parametri nell'environment del docker-compose.yml:

- transparency.google.maps.enabled=true
- transparency.google.maps.key=LA_CHIAVE_DA_PRELEVARE_DAI_SISTEMI_GOOGLE

4.5 Result Service

Result Service è il componente che si occupa di gestire i risultati delle verifiche sulla corrispondenza dei siti degli enti pubblici.

Result Service mantiene nel proprio datastore locale le informazioni relative ai risultati di validazione.

Il codice sorgente è disponibile su github:

- <https://github.com/trasparenzai/result-service>

Nel repository github è compreso anche uno script per la prima installazione del servizio `first-setup.sh`.

In particolare è necessario configurare la sezione della sicurezza.

4.5.1 Sicurezza

Gli endpoint REST di questo servizio sono protetti tramite autenticazione OAuth con Bearer Token. È necessario configurare l'IDP da utilizzare per validare i token OAuth tramite le due proprietà impostabili nel docker-compose.yml come nell'esempio seguente:

- spring.security.oauth2.resourceserver.jwt.issuer-uri=https://dica33.ba.cnr.it/keycloak/
 ↳ realms/trasparenzai
- spring.security.oauth2.resourceserver.jwt.jwk-set-uri=https://dica33.ba.cnr.it/
 ↳ keycloak/realms/trasparenzai/protocol/openid-connect/certs

I valori dei parametri `jwt.issuer-uri` e `jwk-set-uri` sono quelli già descritti nella sezione [Autenticazione](#).

4.6 Result Aggregator Service

Result Aggregator Service è il componente che si occupa di gestire i risultati delle verifiche sulla corrispondenza, aggregando i risultati di validazione con altre informazioni sugli enti pubblici prelevate da altri servizi.

Result Aggregator Service mantiene nel proprio datastore locale le informazioni relative ai risultati di validazione.

Il codice sorgente è disponibile su github:

- <https://github.com/trasparenzai/result-aggregator-service>

Nel repository github è compreso anche uno script per la prima installazione del servizio `first-setup.sh`.

Questo servizio ha due dipendenze per funzionare:

- il [Result Service](#) da cui leggere le info sulle verifiche
- il [Public Sites Service](#) da cui prelevare le info geografiche delle PA

Attenzione: se il `public-site-service` o il `result-service` non sono avviati sullo stesso server tramite docker è necessario configurare l'url a cui rispondono, modificando nel `.env` le variabili d'ambiente `TRANSPARENCY_PUBLIC_SITE_URL` e `TRANSPARENCY_RESULT_SERVICE_URL`.

Per esempio nel `.env`:

```
TRANSPARENCY_PUBLIC_SITE_URL=https://dica33.ba.cnr.it/public-sites-service  
TRANSPARENCY_RESULT_SERVICE_URL=https://dica33.ba.cnr.it/result-service
```

Per configurare il client REST che accede a questi due servizi è necessario configurare nel .env il parametro **OIDC_CLIENT_SECRET**, impostando il valore generato quando si è creato il *Service Account result-aggregator*, vedi [Autenticazione](#).

Inoltre è necessario configurare la sezione della sicurezza.

4.6.1 Sicurezza

Gli endpoint REST di questo servizio sono protetti tramite autenticazione OAuth con Bearer Token. È necessario configurare l'IDP da utilizzare per validare i token OAuth tramite le due proprietà impostabili nel docker-compose.yml come nell'esempio seguente:

```
- spring.security.oauth2.resourceserver.jwt.issuer-uri=https://dica33.ba.cnr.it/keycloak/  
  ↳realms/trasparenzai  
- spring.security.oauth2.resourceserver.jwt.jwk-set-uri=https://dica33.ba.cnr.it/  
  ↳keycloak/realms/trasparenzai/protocol/openid-connect/certs
```

I valori dei parametri *jwt.issuer-uri* e *jwk-set-uri* sono quelli già descritti nella sezione [Autenticazione](#).

4.7 Task Scheduler Service

Task Scheduler Service è il componente che si occupa di avviare alcuni processi eseguiti a intervalli fissi, come per esempio l'**avvio delle scansioni** dei siti del PA per la verifica della corrispondenza dei requisiti e la **cancellazione dei risultati di scansione più vecchi**.

Nell'utilizzo tramite **docker-compose.yml** ricordarsi di impostare nel **.env** la corretta variabile d'ambiente che specifica l'url del config-service da utilizzare e la password per l'autenticazione Basic Auth con il *config-service*:

```
environment:  
  - confighost=${CONFIG_HOST}  
  - spring.security.oauth2.client.registration.oidc.client-secret=${OIDC_CLIENT_SECRET}
```

Vedi [Config Service](#).

Il codice sorgente è disponibile su github:

- <https://github.com/trasparenzai/task-scheduler-service>

Nel repository github è compreso anche uno script per la prima installazione del servizio `first-setup.sh`.

Questo servizio ha tre dipendenze per funzionare:

- il conductor-service tramite cui avviare i flussi di verifica e la cancellazione dei vecchi workflow
- il *Result Service* da cui cancellare i vecchi workflow
- il *Result Aggregator Service* da cui cancellare i vecchi workflow

La configurazione del **conductor-service** per avviare i nuovi flussi e cancellarli quelli vecchi viene letta automaticamente dal [Config Service](#).

Attenzione: è invece importante impostare nel **.env** le URL dei servizi *result-service* e *result-aggregator-service* modificando le variabili d'ambiente **TRANSPARENCY_RESULT_SERVICE_URL** e **TRANSPARENCY_RESULT_AGGRGATOR_SERVICE_URL**:

```
# Configurazione indirizzi dei servizi dove cancellare i risultati del workflow scaduti
- transparency.clients.result-service.url=${TRANSPARENCY_RESULT_SERVICE_URL}
- transparency.clients.result-aggregator-service.url=${TRANSPARENCY_RESULT.Aggregator_
SERVICE_URL}
```

Per configurare il client REST che accede a questi due servizi è necessario configurare nel .env il parametro **OIDC_CLIENT_SECRET**, impostando il valore generato quando si è creato il *Service Account task-scheduler*, vedi [Autenticazione](#):

```
# Client Secret da generare nel Identity Provider e impostare qui
- spring.security.oauth2.client.registration.oidc.client-secret=${OIDC_CLIENT_SECRET}
```

Inoltre è necessario configurare la sezione della sicurezza.

4.7.1 Sicurezza

Gli endpoint REST di questo servizio sono protetti tramite autenticazione OAuth con Bearer Token. È necessario configurare l'IDP da utilizzare per validare i token OAuth tramite le due proprietà impostabili nel docker-compose.yml come nell'esempio seguente:

```
- spring.security.oauth2.resourceserver.jwt.issuer-uri=https://dica33.ba.cnr.it/keycloak/
  ↵realms/trasparenzai
- spring.security.oauth2.resourceserver.jwt.jwk-set-uri=https://dica33.ba.cnr.it/
  ↵keycloak/realms/trasparenzai/protocol/openid-connect/certs
```

I valori dei parametri *jwt.issuer-uri* e *jwk-set-uri* sono quelli già descritti nella sezione [Autenticazione](#).

4.8 Risorse hardware consigliate

I test di funzionamento in produzione hanno evidenziato la necessità di suddividere l'architettura del sistema su almeno 3 distinte virtual machine (VM-A, VM-B, VM-C). In particolare, attese le risorse computazionali necessarie ai vari servizi, si consiglia l'installazione della piattaforma *TrasparenzAI* su almeno tre sistemi separati: VM-A) crawler ; VM-B) coordinamento; VM-C) gestione dei risultati e visualizzazione via Web.

Di seguito le risorse utilizzate per i test di funzionamento della piattaforma *TrasparenzAI*:

Table4.1: Risorse VM-A

Software di virtualizzazione	VMware vSphere 7.5
HW CPU	Intel Xeon Gold 6342
VM Virtual CPU	32 vCPU
VM RAM	64GB
VM HDD sistema operativo	500GB
Sistema operativo VM	Ubuntu Server 24.04
Servizi e componenti	crawler-service, Selenium (Chrome Driver)

Table4.2: Risorse VM-B

Software di virtualizzazione	VMware vSphere 7.5
HW CPU	Intel Xeon CPU E7-4890v2
VM Virtual CPU	32 vCPU
VM RAM	32GB
VM HDD data	2TB
VM HDD sistema operativo	500GB
Sistema operativo VM	Ubuntu Server 24.04
Servizi e componenti	<i>Rule Service, Conductor service</i>

Table4.3: Risorse VM-C

Software di virtualizzazione	VMware vSphere 7.5
HW CPU	Intel Xeon CPU E7-4890v2
VM Virtual CPU	32 vCPU
VM RAM	64GB
VM HDD data	1TB
VM HDD sistema operativo	500GB
Sistema operativo VM	Ubuntu Server 24.04
VM Servizi e componenti	<i>Config Service, Public Sites Service, Result Aggregator Service, Result Service, UI service, Task Scheduler Service, Traefik, Keycloak, Minio</i>

CHAPTER 5

Manuale Amministratore

Questo manuale descrive le funzioni disponibili agli utenti con ruolo «Amministratore» (Administrator) attraverso interfaccia web. L'interfaccia web permette accessi a funzioni differenziate in base al ruolo degli utenti (Public, Power User, Super User, Administrator). Per le informazioni relative all'installazione del software, la configurazione avanzata, il tuning e le scelte di sistema, si rinvia alle sezioni della documentazione che descrivono le funzionalità dei singoli servizi e componenti della piattaforma TrasparenzAI.

5.1 Homepage

La homepage (Fig. 5.1) della piattaforma TrasparenzAI è stata progettata per garantire immediatezza e semplicità nella consultazione dei risultati ottenuti. L'interfaccia mostra chiaramente i dati relativi all'ultima scansione effettuata, fornendo un quadro sintetico del numero di pubbliche amministrazioni analizzate e della conformità riscontrata. È possibile visualizzare la cronologia dettagliata dei controlli precedenti, ciascuno accompagnato da un report scaricabile in formato CSV, e una timeline interattiva che permette di consultare rapidamente l'evoluzione dei controlli nel tempo.

5.1.1 Header (intestazione)

Nella parte superiore della homepage (Fig. 5.2) sono presenti:

1. La barra superiore con il link al protocollo di collaborazione tra Consiglio Nazionale delle Ricerche e Autorità Nazionale Anticorruzione nell'ambito del quale è stato sviluppata la **piattaforma TrasparenzAI** ([Protocollo d'intesa tra l'Autorità Nazionale Anticorruzione e il Consiglio Nazionale delle Ricerche del 7 agosto 2023](#)) e i link di accesso rapido a credits, documentazione, login.
2. Il menu di navigazione principale per l'accesso alle principali funzioni «Cerca amministrazioni», «Mappa delle Amministrazioni», «Regole», «Grafici e Mappe», «Esplora Sezioni». Il Menù è responsive e può essere collassato su dispositivi mobili.

5.1.2 Sezione ultimo controllo

La homepage mostra i principali dati dell'ultimo controllo eseguito con una layout a griglia con card (Fig. 5.3). La prima card illustra i dati raggruppati per stato, la card centrale invece mostra una torta con le percentuali degli stati la cui legenda è esplicitata nell'ultima card a destra.



Figure5.1: Interfaccia Web - Homepage



Figure5.2: Header (intestazione)

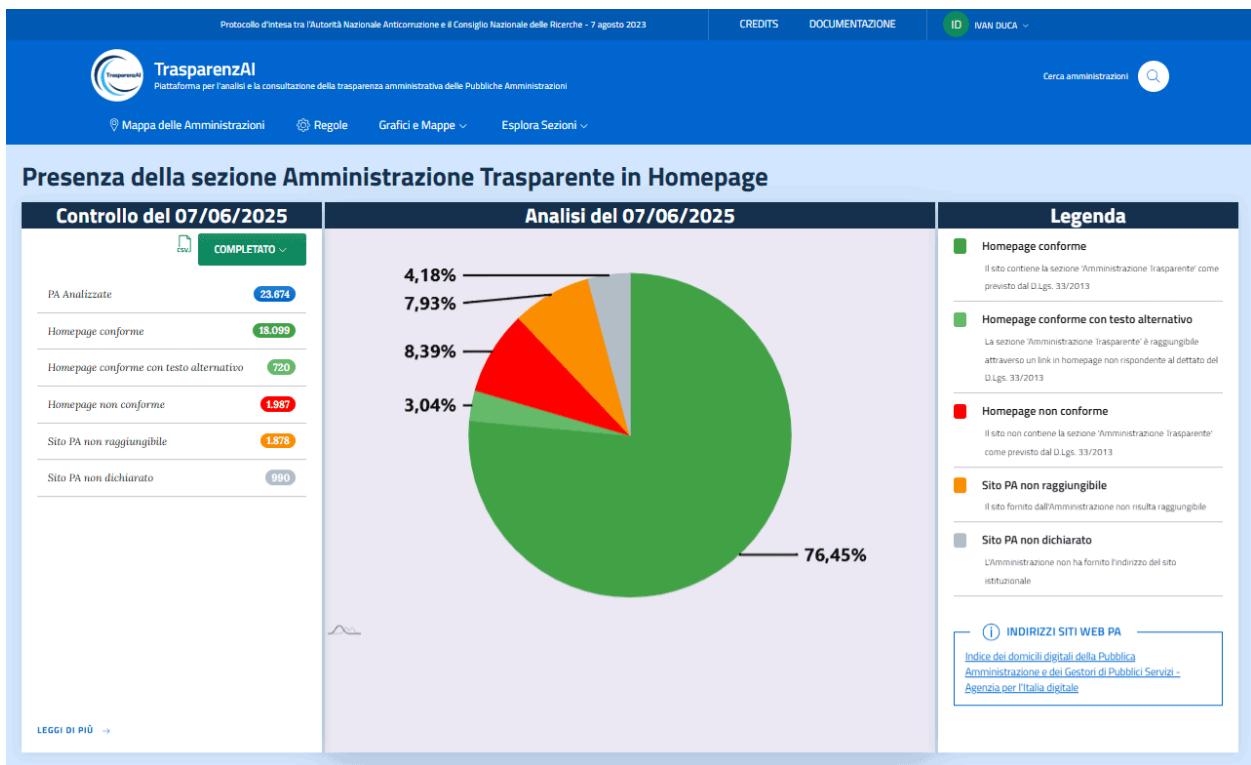


Figure5.3: Sezione ultimo controllo

5.1.3 Sezione cronologia dei controlli

La piattaforma offre un riepilogo visivo delle scansioni precedenti, ciascuna con dati sintetici e la possibilità di esportare i risultati. La sezione è composta da un carousel che permette di scorrere la cronologia dei controlli memorizzati dal sistema. Le informazioni includono data e ora del controllo, numero di amministrazioni analizzate, conformità rilevate. Qualora si abbia il ruolo necessario, è possibile l'esportazione in formato *csv* dei dati. Il link «*LEGGI DI PIÙ*» > mostra la distribuzione geografica dei dati del singolo controllo (Fig. 5.4).



Figure5.4: Sezione cronologia dei controlli

5.1.4 Sezione timeline

La sezione in homepage presenta una timeline verticale delle scansioni con il riepilogo dei risultati ottenuti. Ogni nodo nella linea del tempo consente di consultare nel dettaglio i dati della relativa scansione, fornendo una visione cronologica dell'attività di monitoraggio. Per ogni nodo è anche presente un link attivo alla distribuzione geografica dei dati del singolo controllo (Fig. 5.5).

5.2 Menù «Cerca Amministrazioni»

La piattaforma dispone di una funzione di ricerca avanzata (Fig. 5.6) che permette di individuare specifiche pubbliche amministrazioni utilizzando filtri. Questa funzionalità semplifica significativamente l'individuazione dell'Amministrazione di cui si ricercano le informazioni e i risultati del monitoraggio. Sono presenti filtri per Codice IPA, Denominazione Ente, Codice Fiscale, CATEGORIA (es. Comuni, Istituti scolastici, Ordini professionali), Comune, Provincia, Regione e la selezione del criterio di ordinamento. I risultati della ricerca mostrano le informazioni generali delle Amministrazioni e le funzioni «Controlli», «Storico», «Regole», «Mappe» per l'accesso diretto ai dati dell'Amministrazione selezionata (Fig. 5.7).

5.3 Menù «Mappa delle Amministrazioni»

Il menù «Mappa delle Amministrazioni» accede alla funzione di visualizzazione geografica interattiva delle Amministrazioni (Fig. 5.8). L'utente può così individuare l'Amministrazione che intende visualizzare. Il sistema restituisce le informazioni generali dell'Amministrazione, le sezioni relative alla trasparenza pubblicate sul sito web e le statistiche relative allo stato dei dati rilevati. Se l'utente lo permette, il sistema permette di individuare le Amministrazioni presenti nell'area dell'utente attraverso un meccanismo di geolocalizzazione locale (l'applicazione richiede l'autorizzazione all'utente per rilevare le coordinate geografiche di posizione che rimangono esclusivamente sul dispositivo utente e non vengono acquisite dalla piattaforma).

Utilizzando la rotellina del mouse oppure tramite la funzione di zoom dei sistemi touch screen (mobile e tablet) è



Figure5.5: Sezione timeline

The screenshot shows the TrasparenzAI platform's search interface. At the top, there is a header with the logo, the title "TrasparenzAI", and a search bar labeled "Cerca amministrazioni" with a magnifying glass icon. Below the header, there are several search filters: "Codice IPA", "Denominazione Ente", "Codice Fiscale", "Categoria", "Comune", "Provincia", "Regione", "Ordinamento", and "Presenti 23.667 occorrenze. Mostrate dalla 1 alla 24.". The main area displays a grid of six administrative entities:

- Ordine Medici Veterinari della Provincia di Genova**
 - Codice IPA: **omvpg**
 - Codice Fiscale: **80074470107**
 - Categoria: **Federazioni Nazionali, Ordini, Collegi e Consigli Professionali**
 - Tipologia: **Pubbliche Amministrazioni**
 - Sito istituzionale: http://www.fnovi.it/ordini-provinciali/Genova_id_ordine_prov=35
 - Indirizzo: **Piazza Borgo Pila 40/47 - 16141 - Genova**
 - Regione: **Liguria**
 - Presidente: **Giuseppe Peirano**
 - Pec: ordinevet.ge@pec.fnovi.it
 - Altro: info@ordineveterinariogenova.org
- ISTRUZIONE SECONDARIA SUPERIORE - MACHIAVELLI**
 - Codice IPA: **istsc_fisi00100r**
 - Codice Fiscale: **80024210488**
 - Categoria: **Istituti di Istruzione Statale di Ogni Ordine e Grado**
 - Tipologia: **Pubbliche Amministrazioni**
 - Sito istituzionale: www.licemachiavelli-firenze.gov.it
 - Indirizzo: **Via Santo Spirito,39 - 50125 - Firenze**
 - Regione: **Toscana**
 - Dirigente scolastico: **Francesco Giari**
 - Pec: fisi00100r@pec.istruzione.it
 - Altro: fisi00100r@istruzione.it
- Ordine dei Medici Veterinari della Provincia di Pescara**
 - Codice IPA: **omvpa**
 - Codice Fiscale: **80020880680**
 - Categoria: **Federazioni Nazionali, Ordini, Collegi e Consigli Professionali**
 - Tipologia: **Pubbliche Amministrazioni**
 - Sito istituzionale: www.fnovi.it
 - Indirizzo: **Via Chieti, 5 - 65124 - Pescara**
 - Regione: **Abruzzo**
 - Presidente: **Leonardo Marchegiani**
 - Pec: ordinevet.pe@pec.fnovi.it
 - Altro: ordvetpe@tin.it
- ISTRUZIONE SECONDARIA SUPERIORE - 'LEINAUDI'**
 - Codice IPA: **istsc_cris00600t**
 - Codice Fiscale: **80003440197**
 - Categoria: **Istituti di Istruzione Statale di Ogni Ordine e Grado**
 - Tipologia: **Pubbliche Amministrazioni**
 - Sito istituzionale: <https://www.einaudicremona.edu.it>
- Ordine Dei Medici Veterinari della Provincia di Chieti**
 - Codice IPA: **omvpc**
 - Codice Fiscale: **80003810696**
 - Categoria: **Federazioni Nazionali, Ordini, Collegi e Consigli Professionali**
 - Tipologia: **Pubbliche Amministrazioni**
- Consorzio di Campocatino in Liquidazione**
 - Codice IPA: **cfc**
 - Codice Fiscale: **80004390607**
 - Categoria: **Consorzi tra Amministrazioni Locali**
 - Tipologia: **Pubbliche Amministrazioni**
 - Indirizzo: **Piazza Gramsci 13 - 03100 - Frosinone**

Figure5.6: Menù - Cerca Amministrazioni

The screenshot shows the TrasprenzAI platform interface. At the top, there is a navigation bar with links for 'Mappa delle Amministrazioni', 'Regole', 'Grafici e Mappe', and 'Esplora Sezioni'. On the right side of the header, there are user profile and search options. Below the header, a search bar is labeled 'Filtrala tua ricerca'.

The main content area displays a search result for 'Comune di Napoli'. The result is presented in a card format:

- Denominazione Ente:** Comune di Napoli
- Codice IPA:** c_f839
- Codice Fiscale:** 80014890638
- Categoria:** Comuni e loro Consorzi e Associazioni
- Tipologia:** Pubbliche Amministrazioni
- Sito istituzionale:** www.comune.napoli.it
- Indirizzo:** PALAZZO SAN GIACOMO PIAZZA MUNICIPIO - 80133 - Napoli
- Regione:** Campania
- Sindaco:** Gaetano Manfredi
- Pec:** protocollo@pec.comune.napoli.it
- Pec:** atti.giudiziari@pec.comune.napoli.it
- Pec:** direzione.generale@pec.comune.napoli.it
- Pec:** ragioneria.generale@pec.comune.napoli.it
- Pec:** segretario.generale@pec.comune.napoli.it

Below the card, there are four buttons: 'Controlli', 'Storico', 'Mappa', and 'Regole'. A note at the bottom right of the card states: 'Presenti 1 occorrenze. Mostrate dalla 1 alla 1.'

At the bottom of the page, there is a footer with logos for 'Consiglio Nazionale delle Ricerche' and 'ANAC AUTORITÀ NAZIONALE ANTICORRUZIONE', along with links for 'Note legali' and 'Privacy policy'.

Figure 5.7: Esempio ricerca filtro «Denominazione Ente»



Figure5.8: Menù «Mappa delle Amministrazioni» - immagine generale

possibile ingrandire e ridurre la mappa (Fig. 5.9) fino ad individuare il dettaglio della singola Amministrazione (Fig. 5.10), visualizzarne le informazioni e raggiungere i dati pubblicati della trasparenza amministrativa.

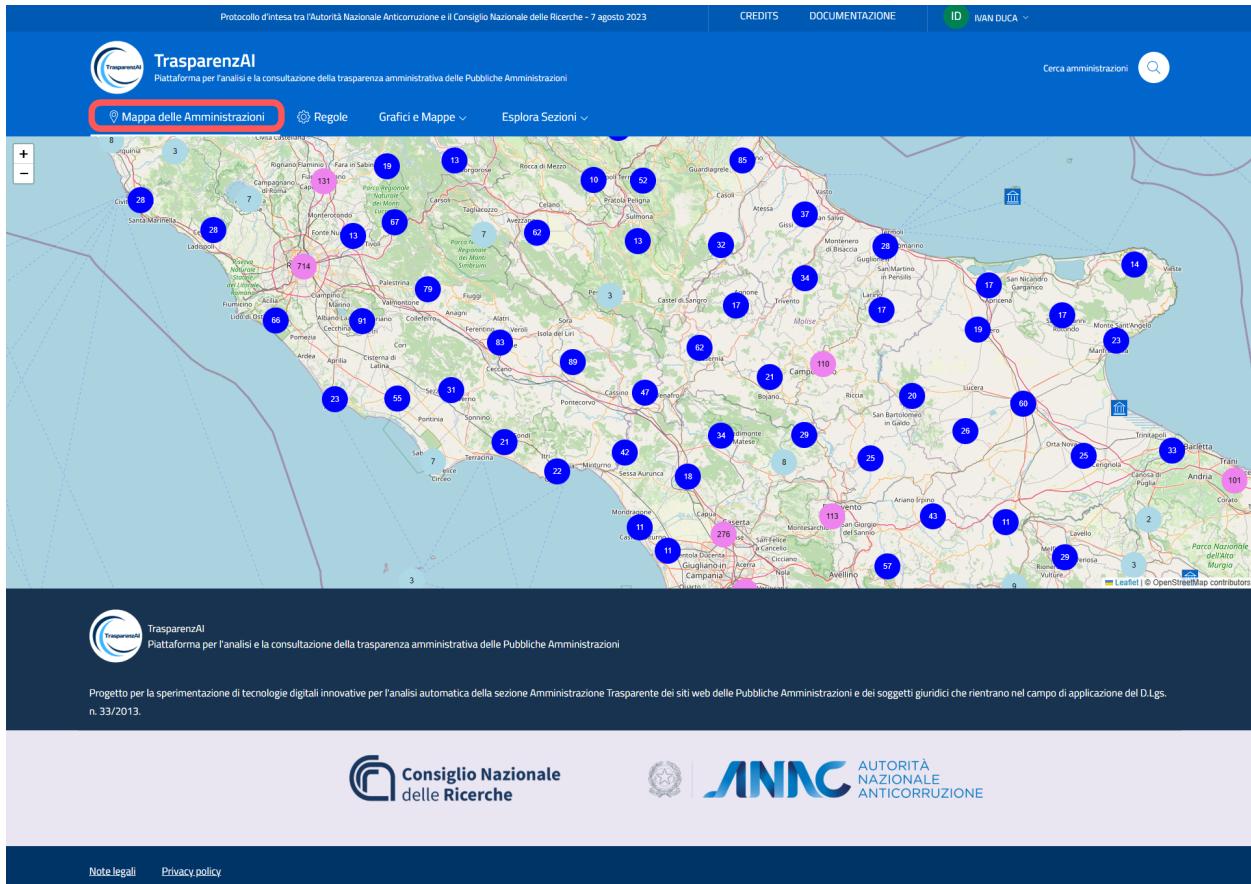


Figure5.9: Menù «Mappa delle Amministrazioni» - zoom 1

5.4 Funzione «Controlli» - Cronologia controlli e Scan-On-Demand

Il tasto “Controlli” attiva la finestra laterale con accesso all’elenco cronologico delle scansioni effettuate (cronologia dei controlli), alle relative informazioni generali (data, orario, tempo di scansione, ecc.), ai risultati delle singole verifiche effettuate. I dati di ogni scansione possono essere esportati dal sistema (per elaborazioni o interscambio) in file di testo con formato CSV. Tramite la medesima finestra, attraverso il tasto «Effettua controllo immediato» (Scan-On-Demand), è possibile avviare una scansione immediata per l’Amministrazione selezionata con la scelta dell’albero di regole da applicare (è possibile effettuare scansioni con alberi differenti per la medesima Amministrazione).

5.5 Funzione «Storico» - Cronologia controlli

Il tasto “Storico” attiva consente l’accesso all’elenco cronologico delle scansioni effettuate (cronologia dei controlli), alle relative informazioni generali (data, orario, tempo di scansione, ecc.), ai risultati delle singole verifiche effettuate. I dati di ogni scansione possono essere esportati dal sistema (per elaborazioni o interscambio) in file di testo con formato CSV. Il colore (verde, arancione, rosso) della sezione intestazione di ogni scansione rappresenta lo stato della rilevazione della sezione «Amministrazione Trasparente» nella Homepage dell’Amministrazione selezionata. Le figure rappresentano lo stato dello storico delle scansioni di Amministrazioni differenti.



Figure5.10: Menù «Mappa delle Amministrazioni» - zoom 2

The screenshot shows the TrasparenzAI platform interface. On the left, there is a sidebar titled "Cronologia dei controlli per: c_f839" listing several completed controls:

- Controllo del 07/06/2025, concluso il 08/06/2025 07:18:48 in 12 ore 18 minuti e 48 secondi
- Controllo del 04/06/2025, concluso il 05/06/2025 07:49:51 in 12 ore 49 minuti e 51 secondi
- Controllo del 01/06/2025, concluso il 01/06/2025 17:16:17 in 12 ore 10 minuti e 57 secondi
- Controllo del 29/05/2025, concluso il 29/05/2025 14:07:46 in 8 ore 20 minuti e 37 secondi
- in 12 ore 18 minuti e 48 secondi
- Controllo del 04/06/2025, concluso il 05/06/2025 07:49:51 in 12 ore 49 minuti e 51 secondi
- Controllo del 01/06/2025, concluso il 01/06/2025 17:16:17 in 12 ore 10 minuti e 57 secondi
- Controllo del 29/05/2025, concluso il 29/05/2025 14:07:46 in 8 ore 20 minuti e 37 secondi
- Controllo del 24/05/2025, concluso il 25/05/2025 17:43:46 in 22 ore 43 minuti e 45 secondi
- Controllo del 22/05/2025, concluso il 22/05/2025 15:17:25

On the right, the main panel displays search filters for "Provincia", "Regione", and "Ordinamento". It shows the results for "COMUNE DI NAPOLI" (Natura: 2430) with one occurrence. At the bottom, there are links for "Storico", "Mappa", and "Regole". A large blue button at the bottom left says "Esegui controllo immediato".

Figure5.11: Funzione «Controlli» e «Effettua controllo immediato»

TrasparenzAI - Piattaforma per l'analisi e la consultazione della trasparenza amministrativa delle Pubbliche Amministrazioni, Release 1.0.0

Protocollo d'intesa tra l'Autorità Nazionale Anticorruzione e il Consiglio Nazionale delle Ricerche - 7 agosto 2023 | CREDITS | DOCUMENTAZIONE | ID: IVAN DUCA |

Cerca amministrazioni 🔍

Mappa delle Amministrazioni | Regole | Grafici e Mappe | Esplora Sezioni | Filtra la tua ricerca ↗

Controllo × Nome della Regola × Regole figlie non trovate × Stato ×

Denominazione Ente Code IPA Nome della Regola Regole figlie non trovate Stato

CSV Denominazione Ente Code IPA Nome della Regola Regole figlie non trovate Stato Ordinamento Creazione descendente Scarica in CSV

Comune di Napoli Acronimo: COMUNE DI NAPOLI Natura: 2430

Code IPA: c_f839 | Codice Fiscale: 80014890638 | Categorie: Comuni e loro Consorzi e Associazioni | Tipologia: Pubbliche Amministrazioni | Sito istituzionale: www.comune.napoli.it

Stato: Conforme | Aggiornato al: 07/06/2023 20:47:51 | Nome della Regola: amministrazione-trasparente | Termine ricercato: Amministrazione Trasparente | URL della Regola: <https://www.comune.napoli.it/amministrazionetrasparente>

Controlli Regole Pagina HTML

Comune di Napoli Acronimo: COMUNE DI NAPOLI Natura: 2430

Code IPA: c_f839 | Codice Fiscale: 80014890638 | Categorie: Comuni e loro Consorzi e Associazioni | Tipologia: Pubbliche Amministrazioni | Sito istituzionale: www.comune.napoli.it

Stato: Conforme | Aggiornato al: 04/06/2023 22:19:18 | Nome della Regola: amministrazione-trasparente | Termine ricercato: Amministrazione Trasparente | URL della Regola: <https://www.comune.napoli.it/amministrazionetrasparente>

Controlli Regole Pagina HTML

Comune di Napoli Acronimo: COMUNE DI NAPOLI Natura: 2430

Code IPA: c_f839 | Codice Fiscale: 80014890638 | Categorie: Comuni e loro Consorzi e Associazioni | Tipologia: Pubbliche Amministrazioni | Sito istituzionale: www.comune.napoli.it

Stato: Conforme | Aggiornato al: 31/05/2023 21:08:46 | Nome della Regola: amministrazione-trasparente | Termine ricercato: Amministrazione Trasparente | URL della Regola: <https://www.comune.napoli.it/amministrazionetrasparente>

Controlli Regole Pagina HTML

Comune di Napoli Acronimo: COMUNE DI NAPOLI Natura: 2430

Code IPA: c_f839 | Codice Fiscale: 80014890638 | Categorie: Comuni e loro Consorzi e Associazioni | Tipologia: Pubbliche Amministrazioni | Sito istituzionale: www.comune.napoli.it

Stato: Conforme | Aggiornato al: 28/05/2025 20:44:36 | Nome della Regola: amministrazione-trasparente | Termine ricercato: Amministrazione Trasparente | URL della Regola: <https://www.comune.napoli.it/amministrazionetrasparente>

Controlli Regole Pagina HTML

Comune di Napoli Acronimo: COMUNE DI NAPOLI Natura: 2430

Code IPA: c_f839 | Codice Fiscale: 80014890638 | Categorie: Comuni e loro Consorzi e Associazioni | Tipologia: Pubbliche Amministrazioni | Sito istituzionale: www.comune.napoli.it

Stato: Conforme | Aggiornato al: 24/05/2025 22:03:11 | Nome della Regola: amministrazione-trasparente | Termine ricercato: Amministrazione Trasparente | URL della Regola: <https://www.comune.napoli.it/amministrazionetrasparente>

Controlli Regole Pagina HTML

Comune di Napoli Acronimo: COMUNE DI NAPOLI Natura: 2430

Code IPA: c_f839 | Codice Fiscale: 80014890638 | Categorie: Comuni e loro Consorzi e Associazioni | Tipologia: Pubbliche Amministrazioni | Sito istituzionale: www.comune.napoli.it

Stato: Conforme | Aggiornato al: 24/05/2025 22:03:11 | Nome della Regola: amministrazione-trasparente | Termine ricercato: Amministrazione Trasparente | URL della Regola: <https://www.comune.napoli.it/amministrazionetrasparente>

Controlli Regole Pagina HTML

Comune di Napoli Acronimo: COMUNE DI NAPOLI Natura: 2430

Code IPA: c_f839 | Codice Fiscale: 80014890638 | Categorie: Comuni e loro Consorzi e Associazioni | Tipologia: Pubbliche Amministrazioni | Sito istituzionale: www.comune.napoli.it

Stato: Conforme | Aggiornato al: 17/05/2025 23:24:20 | Nome della Regola: amministrazione-trasparente | Termine ricercato: Amministrazione Trasparente | URL della Regola: <https://www.comune.napoli.it/amministrazionetrasparente>

Controlli Regole Pagina HTML

Comune di Napoli Acronimo: COMUNE DI NAPOLI Natura: 2430

Code IPA: c_f839 | Codice Fiscale: 80014890638 | Categorie: Comuni e loro Consorzi e Associazioni | Tipologia: Pubbliche Amministrazioni | Sito istituzionale: www.comune.napoli.it

Stato: Conforme | Aggiornato al: 14/05/2025 21:37:46 | Nome della Regola: amministrazione-trasparente | Termine ricercato: Amministrazione Trasparente | URL della Regola: <https://www.comune.napoli.it/amministrazionetrasparente>

Controlli Regole Pagina HTML

Comune di Napoli Acronimo: COMUNE DI NAPOLI Natura: 2430

Code IPA: c_f839 | Codice Fiscale: 80014890638 | Categorie: Comuni e loro Consorzi e Associazioni | Tipologia: Pubbliche Amministrazioni | Sito istituzionale: www.comune.napoli.it

Stato: Conforme | Aggiornato al: 10/05/2025 22:29:58 | Nome della Regola: amministrazione-trasparente | Termine ricercato: Amministrazione Trasparente | URL della Regola: <https://www.comune.napoli.it/amministrazionetrasparente>

Controlli Regole Pagina HTML

Comune di Napoli Acronimo: COMUNE DI NAPOLI Natura: 2430

Code IPA: c_f839 | Codice Fiscale: 80014890638 | Categorie: Comuni e loro Consorzi e Associazioni | Tipologia: Pubbliche Amministrazioni | Sito istituzionale: www.comune.napoli.it

Stato: Conforme | Aggiornato al: 05/03/2025 22:08:15 | Nome della Regola: amministrazione-trasparente | Termine ricercato: Amministrazione Trasparente | URL della Regola: <https://www.comune.napoli.it/amministrazionetrasparente>

Controlli Regole Pagina HTML

Comune di Napoli Acronimo: COMUNE DI NAPOLI Natura: 2430

Code IPA: c_f839 | Codice Fiscale: 80014890638 | Categorie: Comuni e loro Consorzi e Associazioni | Tipologia: Pubbliche Amministrazioni | Sito istituzionale: www.comune.napoli.it

Stato: Conforme | Aggiornato al: 29/12/2024 00:55:33 | Nome della Regola: amministrazione-trasparente | Termine ricercato: Amministrazione Trasparente | URL della Regola: <https://www.comune.napoli.it/amministrazionetrasparente>

Controlli Regole Pagina HTML

Presenti 12 occorrenze. Mostrate dalla 1 alla 12.

TrasparenzAI
Piattaforma per l'analisi e la consultazione della trasparenza amministrativa delle Pubbliche Amministrazioni

Progetto per la sperimentazione di tecnologie digitali innovative per l'analisi automatica della sezione Amministrazione Trasparente dei siti web delle Pubbliche Amministrazioni e dei soggetti giuridici che rientrano nel campo di applicazione del D.Lgs. n. 33/2013.

Consiglio Nazionale delle Ricerche | ANNC AUTORITÀ NAZIONALE ANTICORRUZIONE

Note legali | Privacy policy

Figure 5.12. Funzione «Storico» - esempio 1
5.5. Funzione «Storico» - Cronologia controlli

41

TrasparenzAI - Piattaforma per l'analisi e la consultazione della trasparenza amministrativa delle Pubbliche Amministrazioni, Release 1.0.0

Protocollo d'intesa tra l'Autorità Nazionale Anticorruzione e il Consiglio Nazionale delle Ricerche - 7 luglio 2023 | CREDITS | DOCUMENTAZIONE | ID: MAN DUCA | Cerca amministrazione |

Filtra la tua ricerca

Controllo | Nome della Regola | Regole figlie non trovate | Stato | Denominazione Ente | Codice Fiscale | Ordinamento | Creazione discendente | Scaria in CSV

Denominazione Ente | Codice Fiscale | Categoria | Stato | Controlli | Regole | Pagina HTML | Screenshot |

Denominazione Ente	Codice Fiscale	Categoria	Stato	Controlli	Regole	Pagina HTML	Screenshot
COMUNE DI PONSO Codice IPA: c_g823 Codice Fiscale: 82002950283 Categorie: Comuni e loro Consorzi e Associazioni Tipologia: Pubbliche Amministrazioni Sito istituzionale: www.comune.ponso.it		Natura: 2430	Stato: Conforme Aggiornato al: 07/06/2025 19:13:27 Nome della Regola: amministrazione-trasparente Termino ricercato: Amministrazione Trasparente				
COMUNE DI PONSO Codice IPA: c_g823 Codice Fiscale: 82002950283 Categorie: Comuni e loro Consorzi e Associazioni Tipologia: Pubbliche Amministrazioni Sito istituzionale: www.comune.ponso.it		Natura: 2430	Stato: Non raggiungibile Aggiornato al: 05/06/2025 07:22:12 Nome della Regola: amministrazione-trasparente				
COMUNE DI PONSO Codice IPA: c_g823 Codice Fiscale: 82002950283 Categorie: Comuni e loro Consorzi e Associazioni Tipologia: Pubbliche Amministrazioni Sito istituzionale: www.comune.ponso.it		Natura: 2430	Stato: Non conforme Aggiornato al: 01/06/2025 16:17:59 Nome della Regola: amministrazione-trasparente				
COMUNE DI PONSO Codice IPA: c_g823 Codice Fiscale: 82002950283 Categorie: Comuni e loro Consorzi e Associazioni Tipologia: Pubbliche Amministrazioni Sito istituzionale: www.comune.ponso.it		Natura: 2430	Stato: Conforme Aggiornato al: 29/05/2025 19:16:24 Nome della Regola: amministrazione-trasparente Termino ricercato: Amministrazione Trasparente				
COMUNE DI PONSO Codice IPA: c_g823 Codice Fiscale: 82002950283 Categorie: Comuni e loro Consorzi e Associazioni Tipologia: Pubbliche Amministrazioni Sito istituzionale: www.comune.ponso.it		Natura: 2430	Stato: Non conforme Aggiornato al: 25/05/2025 16:43:27 Nome della Regola: amministrazione-trasparente				
COMUNE DI PONSO Codice IPA: c_g823 Codice Fiscale: 82002950283 Categorie: Comuni e loro Consorzi e Associazioni Tipologia: Pubbliche Amministrazioni Sito istituzionale: www.comune.ponso.it		Natura: 2430	Stato: Non conforme Aggiornato al: 17/05/2025 21:21:06 Nome della Regola: amministrazione-trasparente Termino ricercato: Amministrazione Trasparente				
COMUNE DI PONSO Codice IPA: c_g823 Codice Fiscale: 82002950283 Categorie: Comuni e loro Consorzi e Associazioni Tipologia: Pubbliche Amministrazioni Sito istituzionale: www.comune.ponso.it		Natura: 2430	Stato: Conforme Aggiornato al: 30/04/2025 19:14:24 Nome della Regola: amministrazione-trasparente Termino ricercato: Amministrazione Trasparente				
COMUNE DI PONSO Codice IPA: c_g823 Codice Fiscale: 82002950283 Categorie: Comuni e loro Consorzi e Associazioni Tipologia: Pubbliche Amministrazioni Sito istituzionale: www.comune.ponso.it		Natura: 2430	Stato: Non conforme Aggiornato al: 06/03/2025 05:58:19 Nome della Regola: amministrazione-trasparente				
COMUNE DI PONSO Codice IPA: c_g823 Codice Fiscale: 82002950283 Categorie: Comuni e loro Consorzi e Associazioni Tipologia: Pubbliche Amministrazioni Sito istituzionale: www.comune.ponso.it		Natura: 2430	Stato: Non conforme Aggiornato al: 29/12/2024 09:58:27 Nome della Regola: amministrazione-trasparente				

Presenti 12 occorrenze. Mostrata dalla 1 alla 12.

TrasparenzAI - Piattaforma per l'analisi e la consultazione della trasparenza amministrativa delle Pubbliche Amministrazioni

Progetto per la sperimentazione di tecnologie digitali innovative per l'analisi automatica della sezione Amministrazione Trasparente dei siti web delle Pubbliche Amministrazioni e dei soggetti giuridici che ne entrano nel campo di applicazione del D.Lgs. n. 33/2013.

Consiglio Nazionale delle Ricerche **AUTORITÀ NAZIONALE ANTICORRUZIONE**

[Note legali](#) | [Privacy policy](#)

Figure5.13: Funzione «Storico» - esempio 2

TrasparenzAI - Piattaforma per l'analisi e la consultazione della trasparenza amministrativa delle Pubbliche Amministrazioni, Release 1.0.0

Protocollo d'intesa tra l'Autorità Nazionale Anticorruzione e il Consiglio Nazionale delle Ricerche - 7 agosto 2023

CREDITS DOCUMENTAZIONE ID IVAN DUCA ▾

Cerca amministrazioni

Mappa delle Amministrazioni Regole Grafici e Mappe Esplora Sezioni ▾

Filtra la tua ricerca

Controllo Nome della Regola Amministrazione trasparente Regole figlie non trovate Stato

Denominazione Ente Codice IPA Categorie Ordinamento Creazione discendente Scarica in CSV

c_pof4

COMUNE DI PAGANICO SABINO
Codice IPA: c_pof4
Codice Fiscale: 00113470579
Categoria: Comuni e loro Consorzi e Associazioni
Tipologia: Pubbliche Amministrazioni
Sito istituzionale: comune.paganicosabino.it
Natura: 2430
Stato: Non conforme
Aggiornato al: 08/06/2025 04:54:32
Nome della Regola: amministrazione-trasparente

COMUNE DI PAGANICO SABINO
Codice IPA: c_pof4
Codice Fiscale: 00113470579
Categoria: Comuni e loro Consorzi e Associazioni
Tipologia: Pubbliche Amministrazioni
Sito istituzionale: comune.paganicosabino.it
Natura: 2430
Stato: Non conforme
Aggiornato al: 05/06/2025 06:28:56
Nome della Regola: amministrazione-trasparente

COMUNE DI PAGANICO SABINO
Codice IPA: c_pof4
Codice Fiscale: 00113470579
Categoria: Comuni e loro Consorzi e Associazioni
Tipologia: Pubbliche Amministrazioni
Sito istituzionale: comune.paganicosabino.it
Natura: 2430
Stato: Non conforme
Aggiornato al: 01/06/2025 14:56:40
Nome della Regola: amministrazione-trasparente

COMUNE DI PAGANICO SABINO
Codice IPA: c_pof4
Codice Fiscale: 00113470579
Categoria: Comuni e loro Consorzi e Associazioni
Tipologia: Pubbliche Amministrazioni
Sito istituzionale: comune.paganicosabino.it
Natura: 2430
Stato: Non conforme
Aggiornato al: 29/05/2025 11:19:19
Nome della Regola: amministrazione-trasparente

COMUNE DI PAGANICO SABINO
Codice IPA: c_pof4
Codice Fiscale: 00113470579
Categoria: Comuni e loro Consorzi e Associazioni
Tipologia: Pubbliche Amministrazioni
Sito istituzionale: comune.paganicosabino.it
Natura: 2430
Stato: Non conforme
Aggiornato al: 18/05/2025 10:52:48
Nome della Regola: amministrazione-trasparente

COMUNE DI PAGANICO SABINO
Codice IPA: c_pof4
Codice Fiscale: 00113470579
Categoria: Comuni e loro Consorzi e Associazioni
Tipologia: Pubbliche Amministrazioni
Sito istituzionale: comune.paganicosabino.it
Natura: 2430
Stato: Non conforme
Aggiornato al: 01/05/2025 04:22:28
Nome della Regola: amministrazione-trasparente

COMUNE DI PAGANICO SABINO
Codice IPA: c_pof4
Codice Fiscale: 00113470579
Categoria: Comuni e loro Consorzi e Associazioni
Tipologia: Pubbliche Amministrazioni
Sito istituzionale: comune.paganicosabino.it
Natura: 2430
Stato: Non conforme
Aggiornato al: 06/03/2025 05:12:37
Nome della Regola: amministrazione-trasparente

COMUNE DI PAGANICO SABINO
Codice IPA: c_pof4
Codice Fiscale: 00113470579
Categoria: Comuni e loro Consorzi e Associazioni
Tipologia: Pubbliche Amministrazioni
Sito istituzionale: comune.paganicosabino.it
Natura: 2430
Stato: Non conforme
Aggiornato al: 29/12/2024 08:47:53
Nome della Regola: amministrazione-trasparente

Presenti: 12 occorrenze. Mostrate dalla 1 alla 12.

TrasparenzAI
Plattaforma per l'analisi e la consultazione della trasparenza amministrativa delle Pubbliche Amministrazioni

Progetto per la sperimentazione di tecnologie digitali innovative per l'analisi automatica della sezione Amministrazione Trasparente dei siti web delle Pubbliche Amministrazioni e dei soggetti giuridici che rientrano nel campo di applicazione del D.Lgs. n. 33/2013.

Consiglio Nazionale delle Ricerche ANAC AUTORITÀ NAZIONALE ANTICORRUZIONE

Note legali Privacy policy

Figure5.14: Funzione «Storico» - esempio 3

5.5. Funzione «Storico» - Cronologia controlli

43

5.6 Menù «Regole»

La sezione dedicata agli alberi delle regole consente agli utenti di visualizzare e interagire direttamente con la struttura normativa prevista da ANAC. Gli alberi delle regole sono navigabili tramite comandi intuitivi quali zoom, rotazione e adattamento alla finestra.

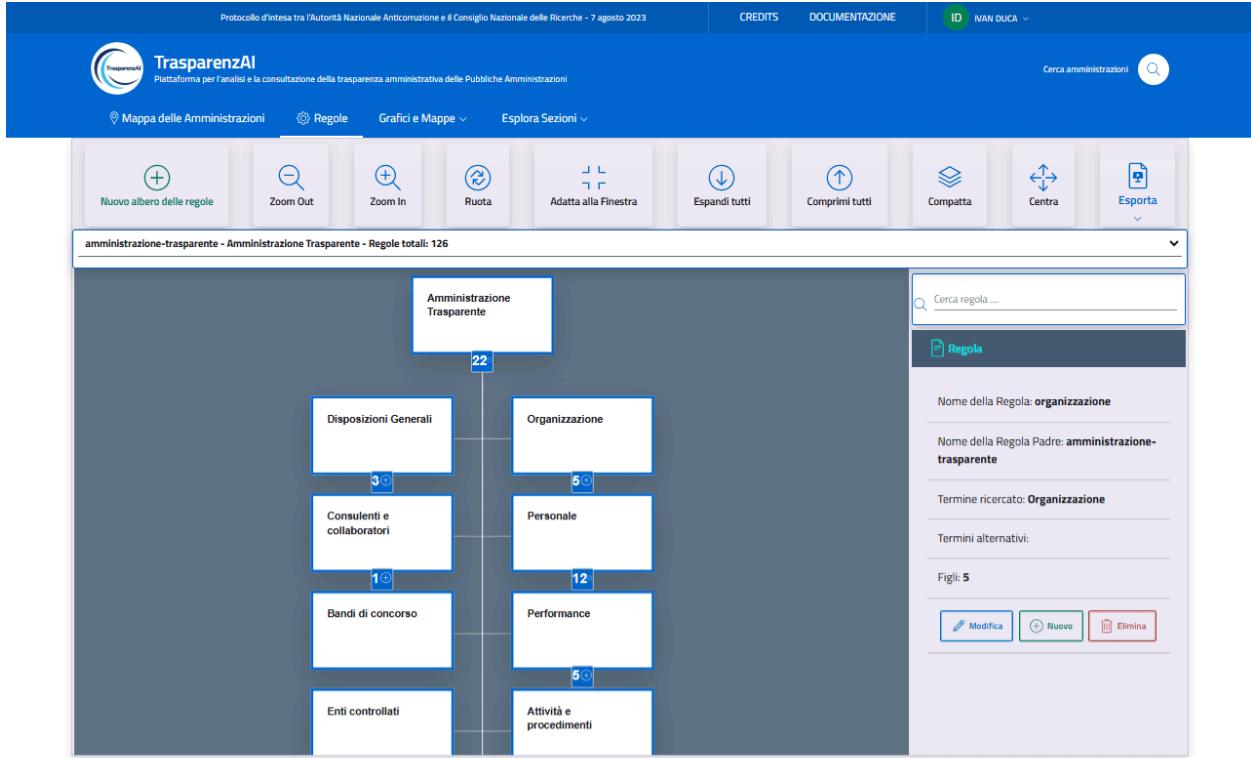


Figure5.15: Menù «Regole»

Il sistema consente di selezionare tra diversi alberi di regole già predisposti o di costruirne di nuovi.

Il tasto «Nuovo albero delle regole» permette di creare nuovi alberi delle regole da applicare a scansioni generali o controlli specifici per singola Amministrazioni o classi di Amministrazioni.

Per ogni elemento dell'albero selezionato, sono presenti i tasti «Modifica», «Nuovo» ed «Elimina». Il tasto «Modifica» permette di modificare i termini da ricercare per ogni regola definendo un elenco di termini che la piattaforma deve considerare equivalenti al termine ricercato nel corso delle scansioni a cui si applica l'albero (esempio termine ricercato «Amministrazione Trasparente», termini alternativi «Società Trasparente», «Amm. Trasparente»). Tale funzionalità, aumenta il grado di tolleranza della piattaforma e permette controlli con gradi di libertà configurabili, adattando la ricerca alla terminologia effettivamente utilizzata nei siti delle Amministrazioni. Il tasto «Nuovo» attiva la funzione di creazione di una nuova regola-figlia della regola attuale (sotto-sezione). Il tasto «Elimina» cancella definitivamente la regola selezionata.

5.7 Menù «Grafico per regola»

La voce di menù «Grafico per regola» (Fig. 5.19) permette di accedere alla rappresentazione grafica dei risultati generati da ogni scansione.

Selezionando una sezione dell'albero delle regole, vengono visualizzati i dati in percentuale e numerici (muovendo il cursore del mouse sulle sezioni dei grafici a torta) relativi alla scansione in esame.

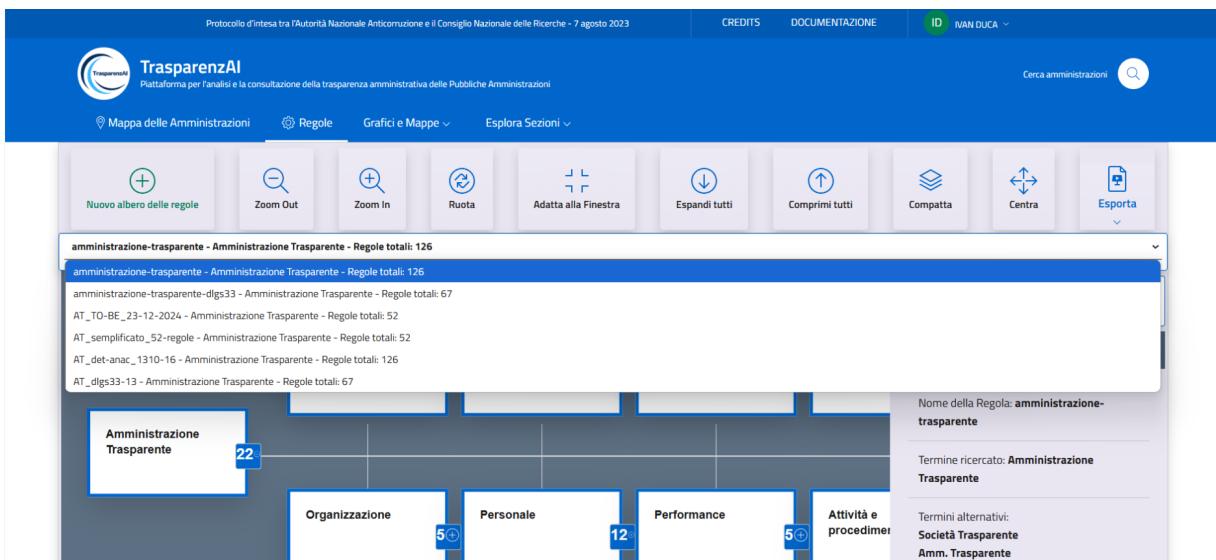


Figure5.16: Selezione albero delle regole per scansione

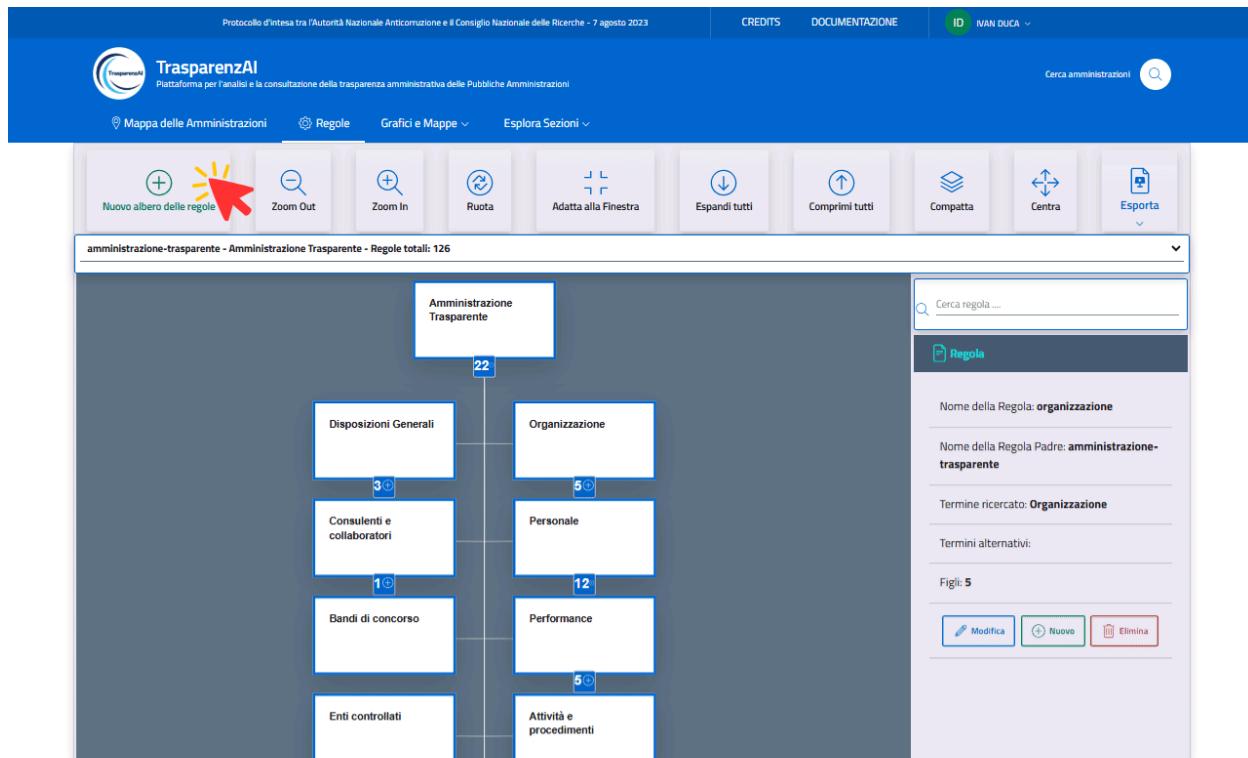


Figure5.17: Creazione di un nuovo albero delle regole

The screenshot shows the 'Regole' (Rules) section of the TrasparenzAI platform. On the left, a tree diagram displays various categories of regulations. The root node is 'Amministrazione Trasparente' (22 items). It branches into 'Disposizioni Generali', 'Organizzazione', 'Personale', 'Performance', and 'Attività e procedimenti'. These further branch into more specific categories like 'Consulenti e collaboratori' (1 item), 'Bandi di concorso' (12 items), and 'Enti controllati'. On the right, a sidebar titled 'Regola' provides search and action functions. A red circle highlights the 'Modifica' (Modify), 'Nuovo' (New), and 'Elimina' (Delete) buttons at the bottom of this sidebar.

Figure5.18: Tasti per la modifica, la creazione e l'eliminazione di una regola

The screenshot shows a dashboard from the TrasparenzAI platform. At the top, there's a banner with the text 'Presenza della sezione Amministrazione Trasparente in Homepage'. Below it, a pie chart titled 'Analisi del 07/06/2025' shows the distribution of website compliance status. The chart is divided into four segments: green (Homepage conforme), orange (Homepage conforme con testo alternativo), red (Homepage non conforme), and grey (Sito PA non raggiungibile). To the left, a table titled 'Controllo del 07/06/2025' provides a detailed breakdown of analyzed sites. A legend on the right explains the colors used in the charts.

Categoria	Conteggio
PA Analizzate	23.674
Homepage conforme	15.099
Homepage conforme con testo alternativo	720
Homepage non conforme	1.987
Sito PA non raggiungibile	1.878
Sito PA non dichiarato	990

Figure5.19: Menù «Grafico per regola»

La Fig. 5.20 mostra la visualizzazione dei risultati relativi alla sezione principale Amministrazione Trasparente per la scansione effettuata il 07/06/2025.

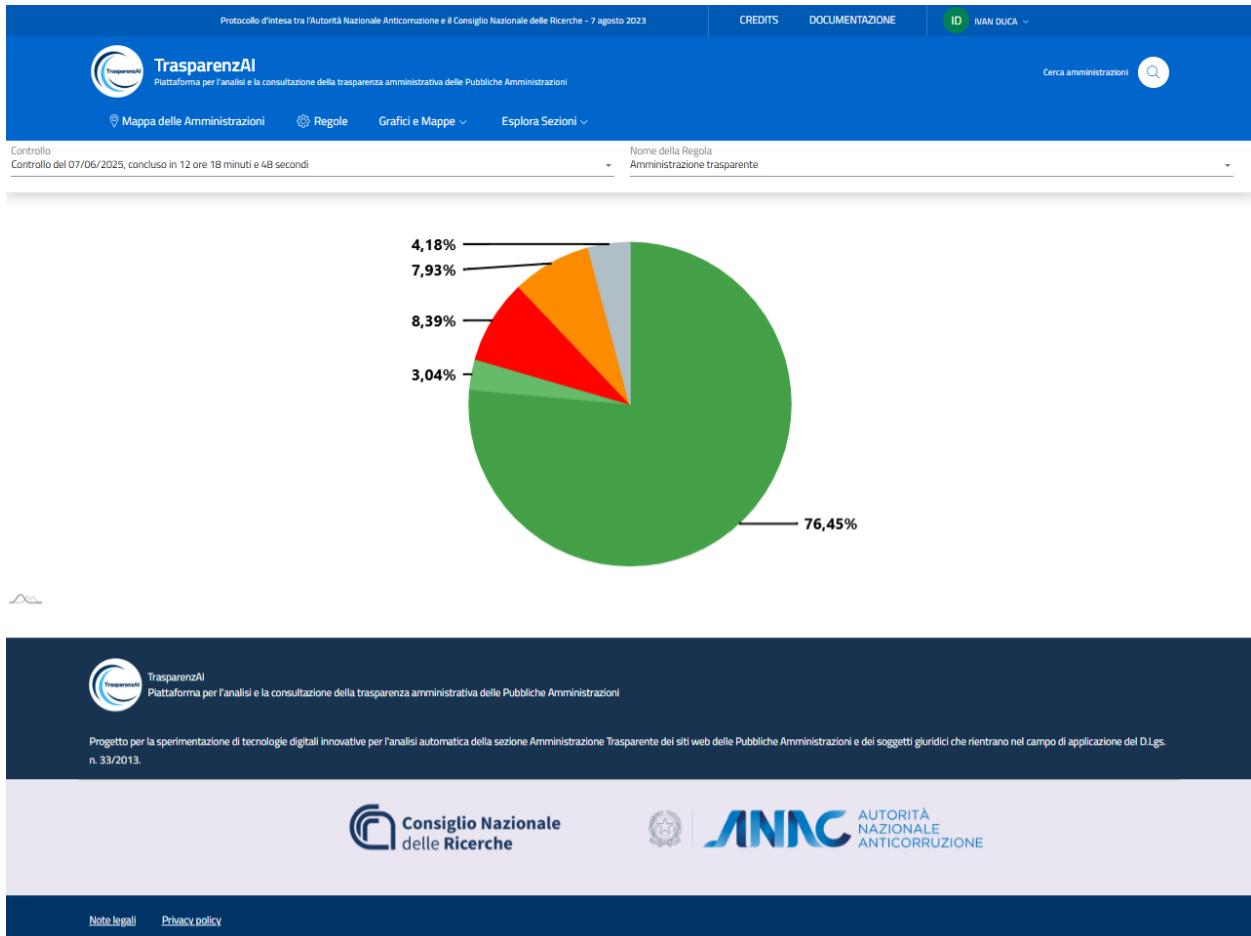


Figure5.20: Menù «Grafico per regola» - Regola «Amministrazione Trasparente» (radice albero)

Il menù a tendina «Nome della Regola» (Fig. 5.21) permette la selezione dei dati di scansione relativi alle varie sezioni dell'albero delle regole.

Nelle figure Fig. 5.22 e Fig. 5.23 sono mostrati i risultati delle sezioni «Organizzazione» e «Tassi di assenza».

La visualizzazione congiunta del livello selezionato (a destra) e del livello superiore (a sinistra) dell'albero delle regole, permette la visualizzazione correlata delle sotto-sezioni con le relative sezioni superiori che le contengono (sezione padre).

5.8 Menù «Mappa per regola»

La voce di menù «Mappa per regola» (Fig. 5.24) permette di accedere alla rappresentazione grafica geolocalizzata dei risultati generati da ogni scansione. Il menù a tendina «Nome della Regola» permette la selezione della sezione dell'albero «Amministrazione Trasparente» di cui si vuole mostrare lo stato su scala geografica. Attraverso la funzione di zoom è possibile definire la dimensione spaziale da visualizzare ed i relativi dati aggregati per area (macro-cluster spaziali). La funzione di zoom permette l'individuazione della singola Amministrazione per visualizzarne i risultati della scansione.

La Fig. 5.25 mostra lo zoom relativo alle macro-area italiane. I grafici sono riferiti alla scansione effettuata il 07/06/2025 per la sezione principale «Amministrazione Trasparente».

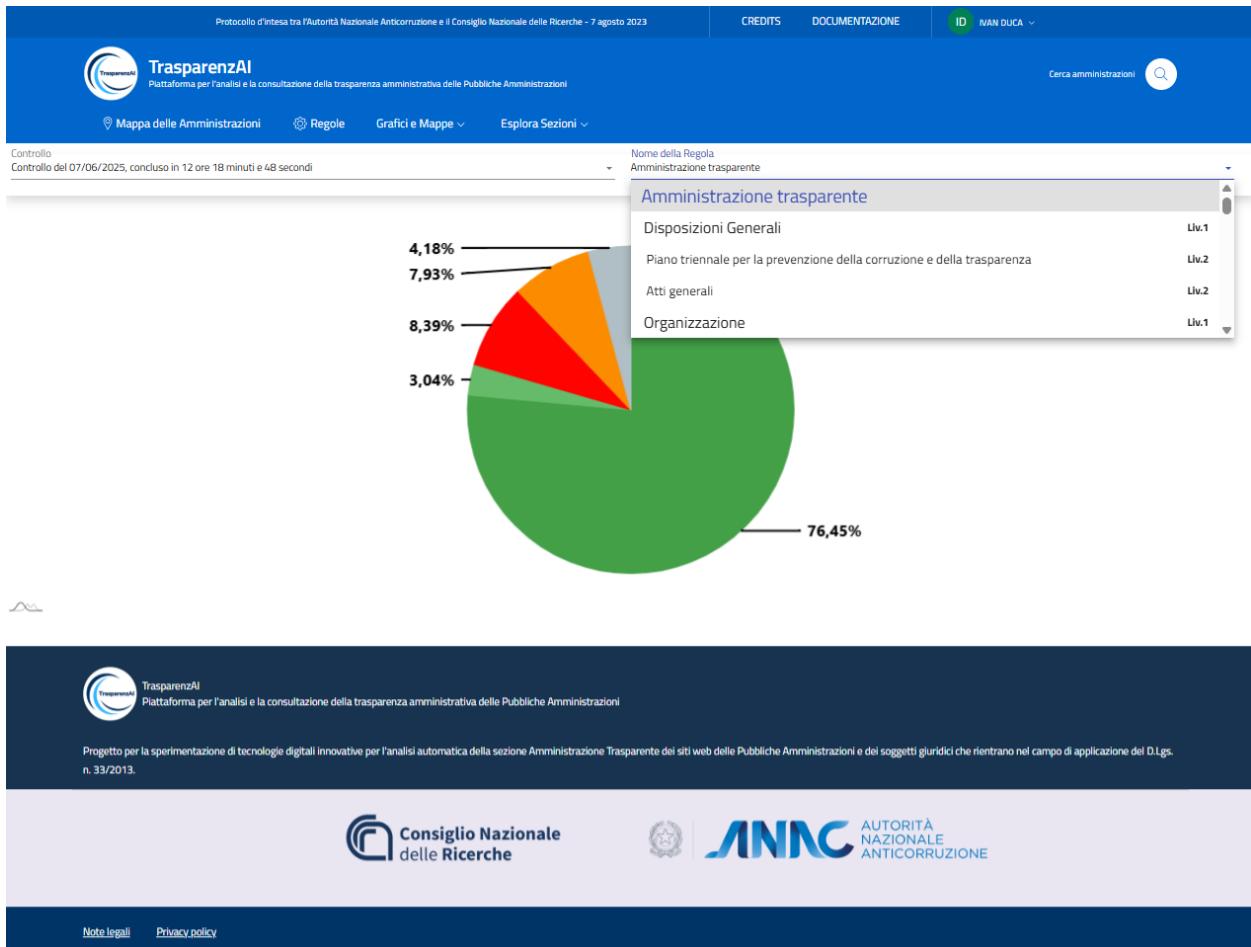


Figure5.21: Menù «Grafico per regola» - selezione «Nome della Regola» da visualizzare

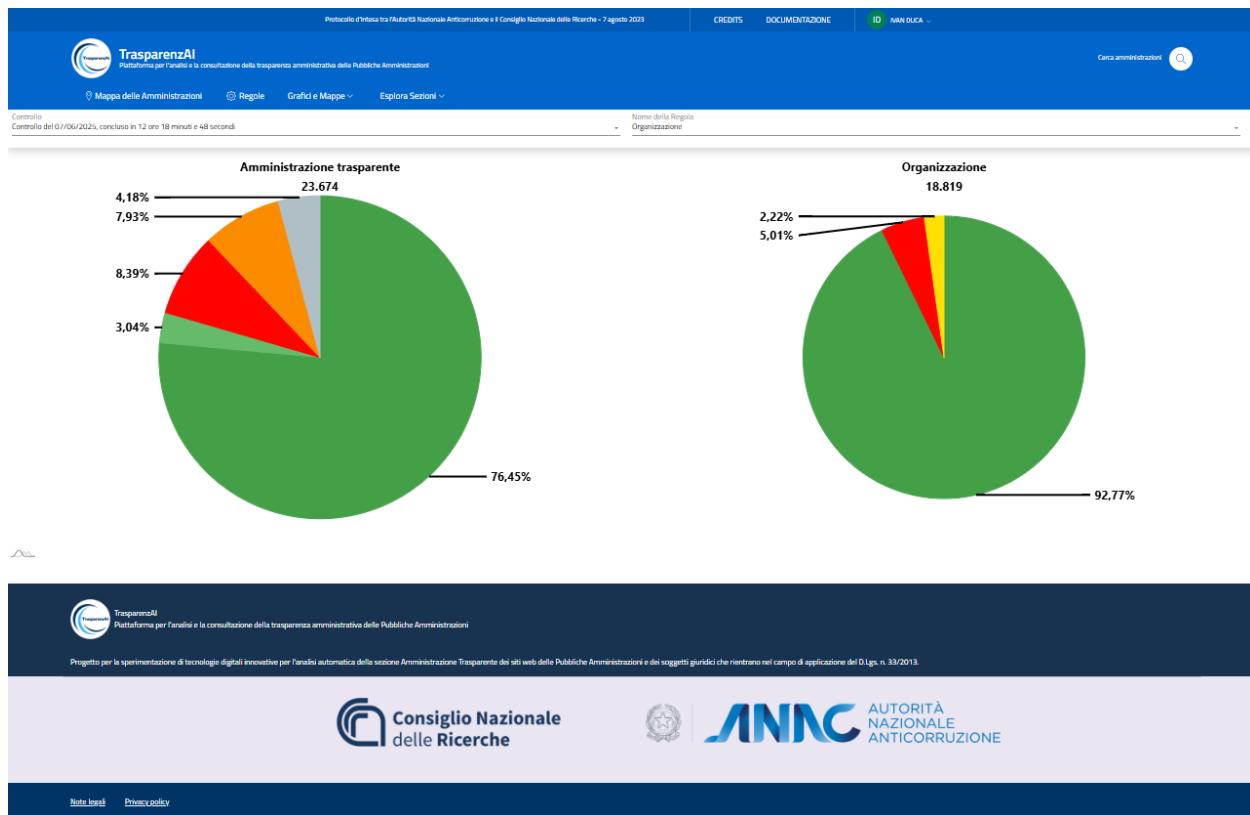


Figure5.22: Menù «Grafico per regola» - Regola «Organizzazione»

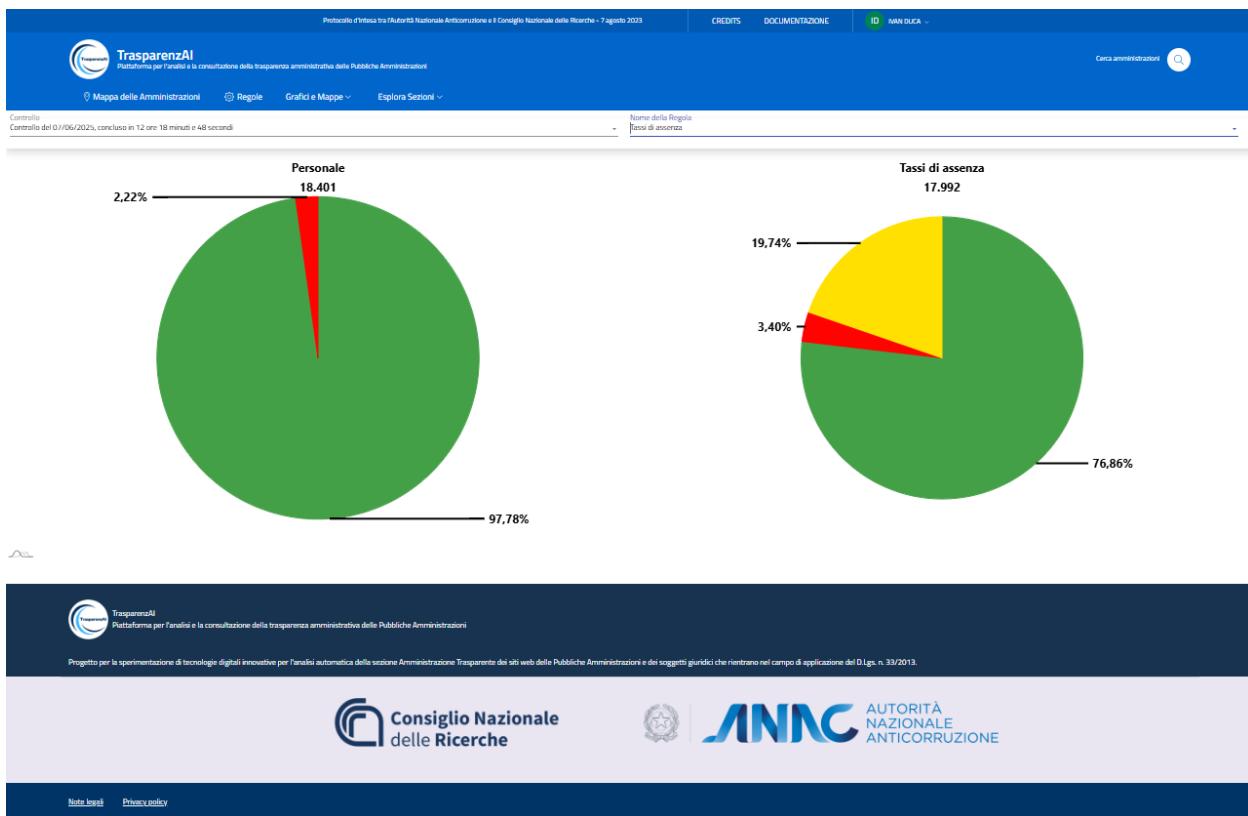


Figure5.23: Menù «Grafico per regola» - Regola «Tasso di assenza»

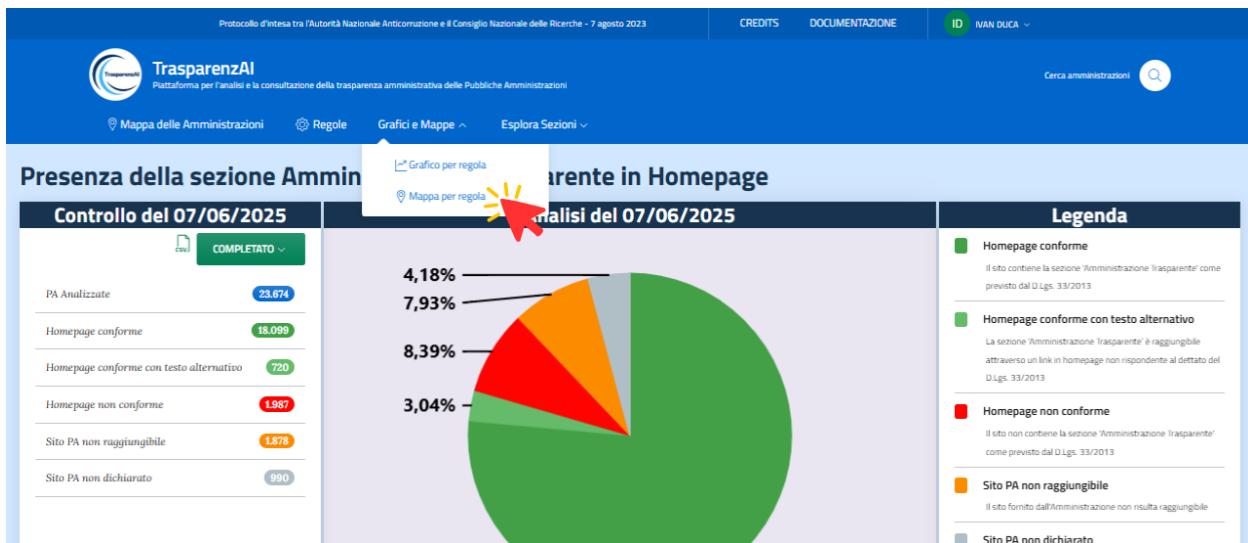


Figure5.24: Menù «Mappa per regola»

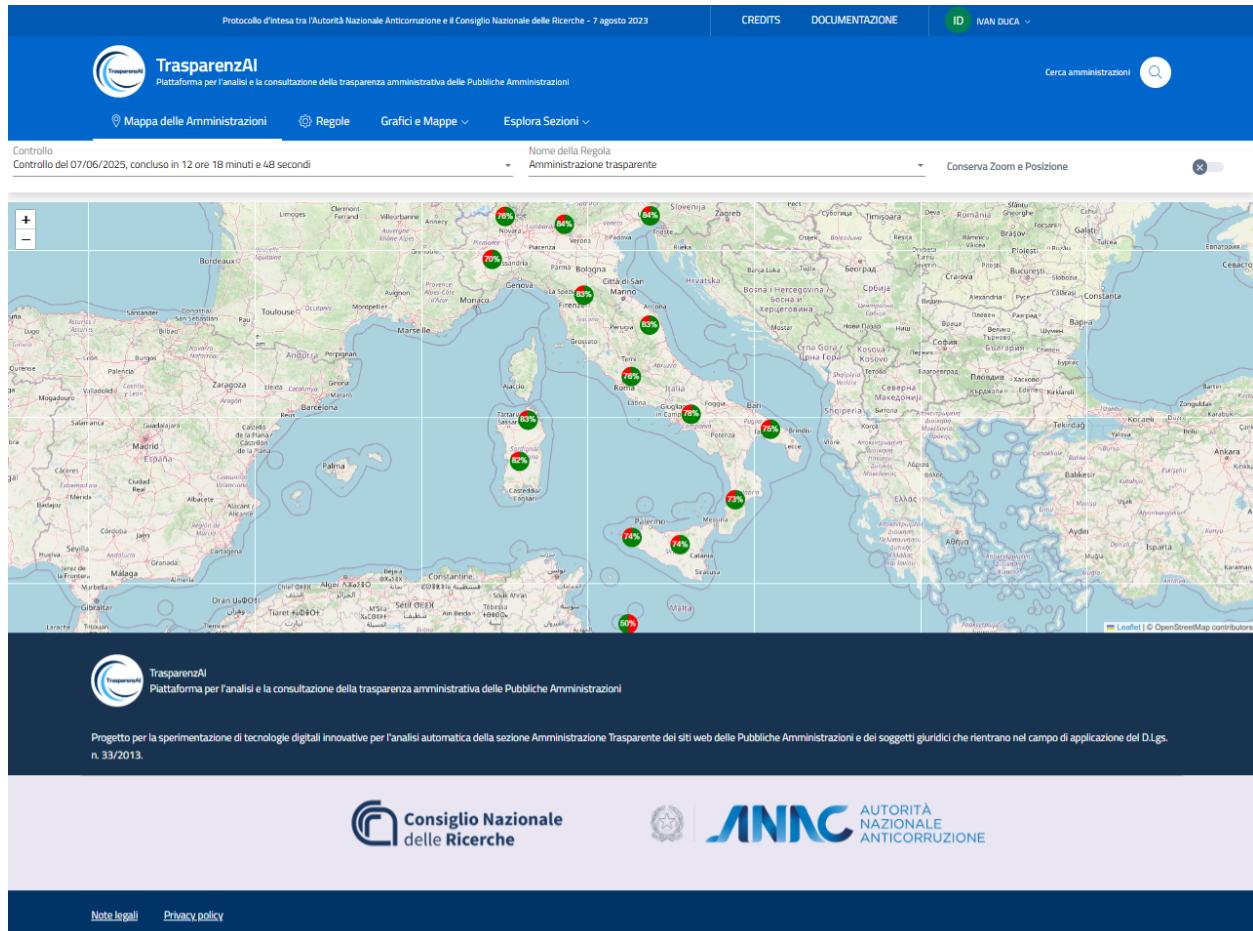


Figure5.25: Menù «Mappa per regola» - Grafici per macro aree zoom 1

Il menù a tendina «Nome della Regola» (Fig. 5.26) permette la selezione dei dati di scansione relativi alle varie sezioni dell'albero delle regole.



Figure 5.26: Menù «Mappa per regola» - selezione «Nome della Regola»

Le figure Fig. 5.27 e Fig. 5.28 mostrano visualizzazioni della sezione principale «Amministrazione Trasparente» con livelli di zoom differenti.

La Fig. 5.29 mostra il livello di zoom massimo con identificazione della singola Amministrazione (esempio: Agenzia Nazionale di Valutazione del Sistema Universitario e della Ricerca). Cliccando sulla denominazione dell'Amministrazione è possibile accedere alla funzione di analisi delle singole regole, per la verifica di tutte le sezioni Amministrazione Trasparente. Per ogni sezione è fornito il link di accesso diretto alle relative pagine informative.

5.9 Menù «Esplora Sezioni»

La voce di menù «Esplora Sezioni» permette di accedere alle funzioni di analisi filtrata dei dati. Il menù permette di accedere alla ricerca con filtro relativa ai controlli effettuati durante le scansioni temporali. E' possibile scegliere la scansione temporale, il nome della regola da verificare, lo stato, la denominazione dell'Amministrazione, il codice fiscale dell'Amministrazione, il codice IPA, la categoria. Tutti i dati sono esportabili con file di interscambio in formato CSV.

La prima voce di menù «Tutte le Regole» permette la ricerca con selezione interamente configurabile dall'utente. Le ulteriori voci di menù offrono ricerche predefinite per specifiche sezioni: Amministrazione Trasparente, Patrimonio immobiliare, Canoni di locazione o affitto, Bandi di gara e contratti, Bandi di concorso, Prevenzione della corruzione, Indicatore di tempestività dei pagamenti, Bilanci.

Le figure di seguito mostrano un esempio di selezione relative alla sezione «Prevenzione della corruzione» con tabelle relative a tutte la Amministrazioni ed alle sole Amministrazioni per le quali la sezione non è stata trovata sul sito web istituzionale.

5.10 Menù «Accesso area riservata»

Dalla barra di menù presente nel header della home page è consentito l'accesso riservato con ruolo «Amministratore» (Administrator) selezionano il menù «Accesso all'area riservata» (ultimo elemento a destra). Inserite le credenziali, si avrà accesso tutte le funzioni della piattaforma accessibili attraverso l'interfaccia web e al sotto-menù «Configurazione» che permette di parametrizzare e personalizzare le principali funzioni della piattaforma stessa.



Figure5.27: Menù «Mappa per regola» - Grafici per macro aree zoom 2

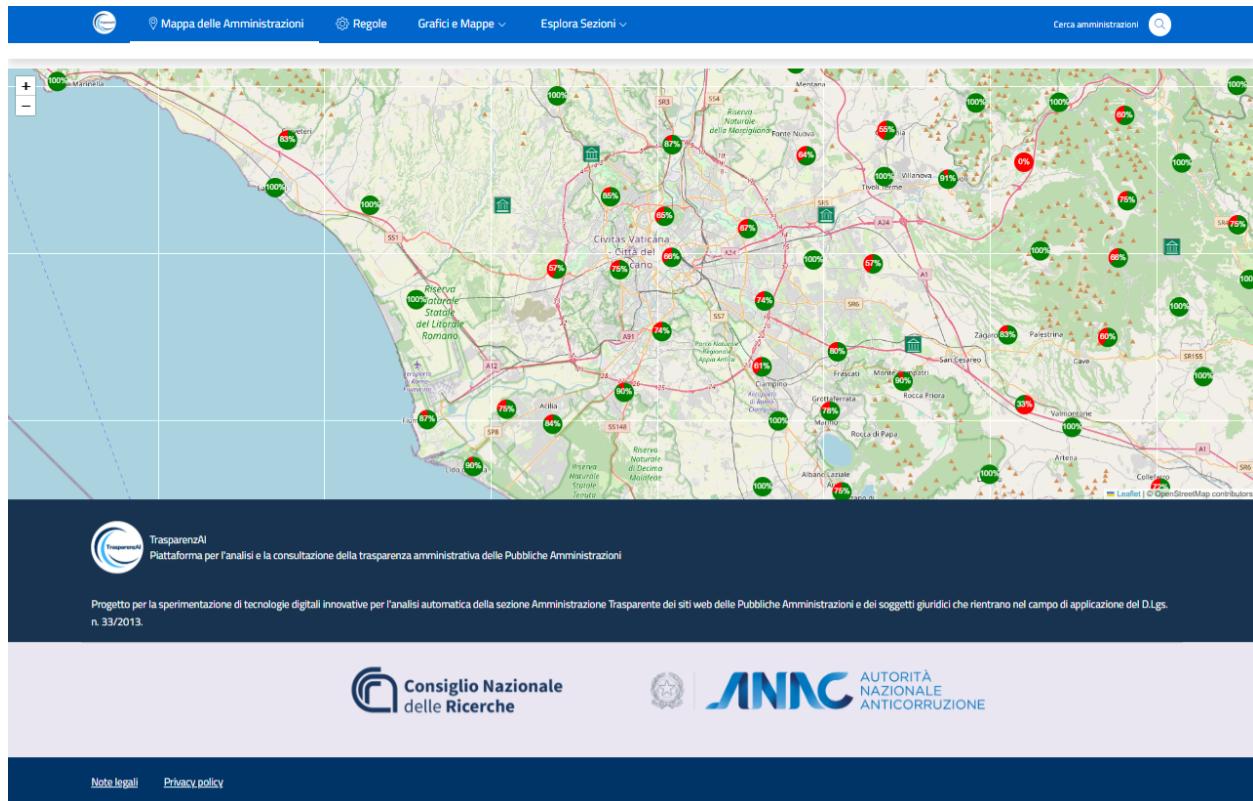


Figure5.28: Menù «Mappa per regola» - Grafici per macro aree zoom 3

Attraverso il sotto-menù «Configurazione» si accede alla pagina di configurazione che permette di definire il calendario di scansione dei siti; i parametri di controllo per le scansioni (alberi delle regole, ,screenshots, timeouts, storico delle scansioni da memorizzare, ecc.); la personalizzazione dei colori. I valori dei parametri sono quelli già descritti nella sezione configurazione del Config Service.

Figure5.29: Menù «Mappa per regola» - singola Amministrazione

Categoria	COUNT
Homepage conforme	15.099
Homepage conforme con testo alternativo	720
Homepage non conforme	1.987
Sito PA non raggiungibile	1.878
Sito PA non dichiarato	990

Figure5.30: Menù «Esplora Sezioni»

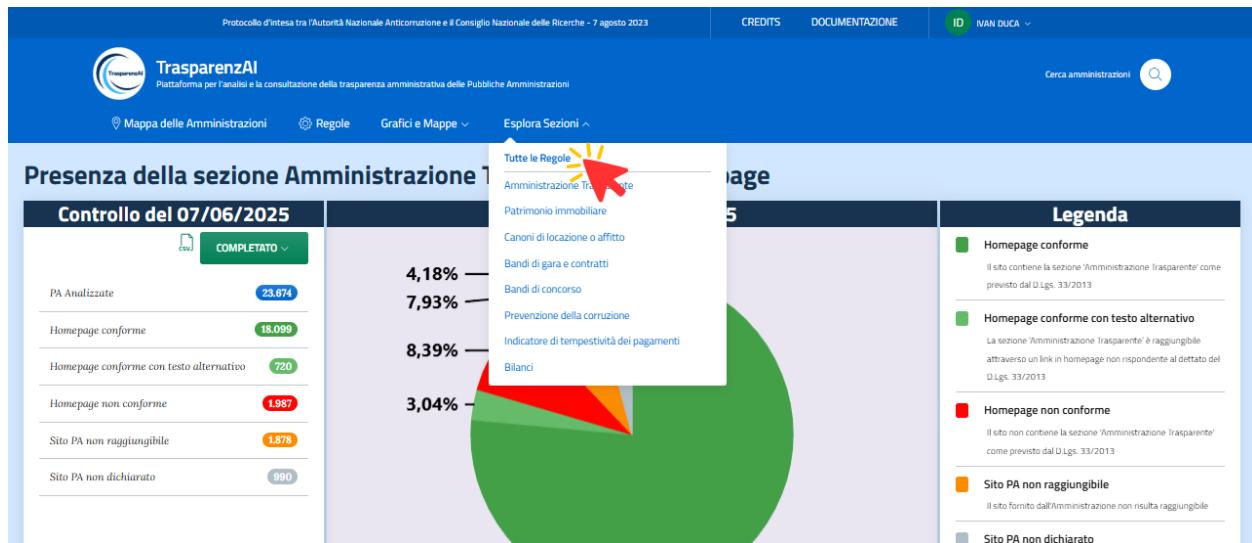


Figure5.31: Menù «Esplora Sezioni» - Filtro «Tutte le Regole»

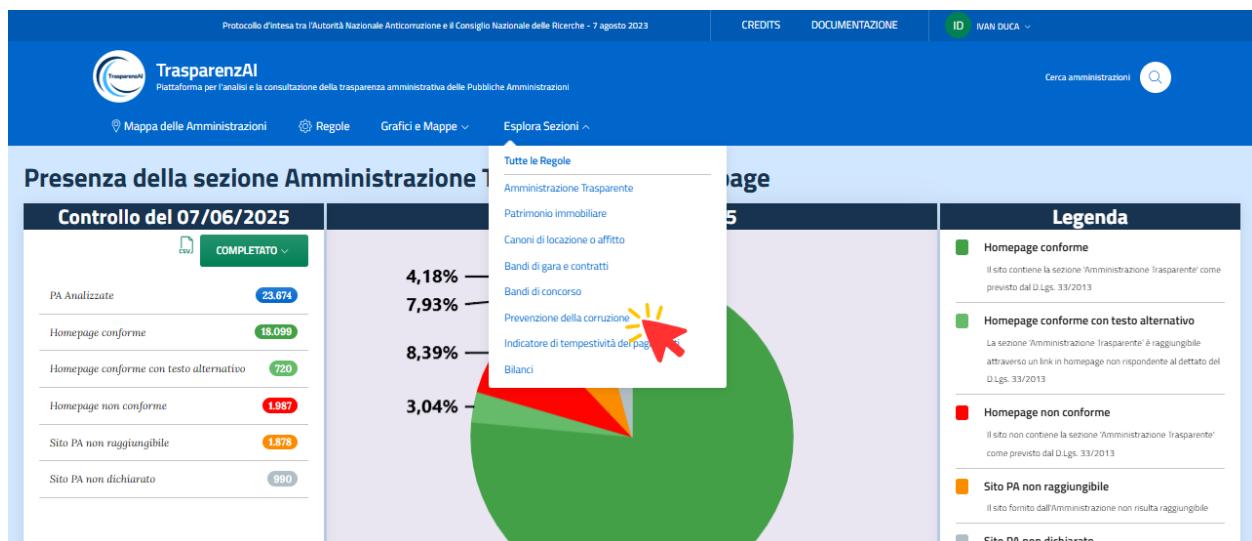


Figure5.32: Menù «Esplora Sezioni» - Filtro «Prevenzione della corruzione»

The screenshot shows the TrasparenzAI platform interface. At the top, there is a navigation bar with links for 'Protocollo d'intesa tra l'Autorità Nazionale Anticorruzione e il Consiglio Nazionale delle Ricerche - 7 agosto 2023', 'CREDITS', 'DOCUMENTAZIONE', 'ID IVAN DUCA', and a search bar labeled 'Cerca amministrazioni' with a magnifying glass icon.

The main area is titled 'Filtro la tua ricerca' (Filter your search). It includes several search filters: 'Controllo' (set to 'Completo'), 'Nome della Regola' (set to 'Prevenzione della Corruzione'), 'Regole figlie non trovate', 'Stato' (set to 'Conforme'), 'Denominazione Ente' (dropdown menu), 'Codice IPA' (dropdown menu), 'Categoria' (dropdown menu), 'Codice Fiscale' (dropdown menu), and 'Ordinamento' (dropdown menu). A 'Scarpa in CSV' button is also present.

The search results are displayed in a grid format. Each result card contains the following information:

- Ente:** Nome della Regola, Codice IPA, Codice Fiscale, Categorie, Tipologia, Sito istituzionale, Stato, Aggiornato al, Nome della Regola, Termine ricercato, URL della Regola, and Links for 'Controlli', 'Regole', 'Storico', and 'Pagina HTML'.
- Example 1:** COMUNE DI INVERNO E MONTELEONE (c_e310, 00484170188, Comuni e loro Consorzi e Associazioni, Pubbliche Amministrazioni, www.comune.invernoemonteleone.pv.it).
- Example 2:** COMUNE DI PRAGELATO (c_g973, 85003290013, Comuni e loro Consorzi e Associazioni, Pubbliche Amministrazioni, www.comune.pragelato.to.it).
- Example 3:** COMUNE DI MONTIANO (c_f668, B1000990408, Comuni e loro Consorzi e Associazioni, Pubbliche Amministrazioni, www.comune.montiano.fc.it).
- Example 4:** ISTITUTO COMPRENSIVO - I.C. MICHELANGELO BUONARROTI (c_stsc_palc87100x, 80026500829, Istituti di Istruzione Statale di Ogni Ordine e Grado, Pubbliche Amministrazioni, www.istitutobuonarroti.edu.it).
- Example 5:** ISTRUZIONE SECONDARIA SUPERIORE - "LEINAUDI" (c_stsc_cris00600t, 800034040197, Istituti di Istruzione Statale di Ogni Ordine e Grado, Pubbliche Amministrazioni, www.einaudicremona.edu.it).
- Example 6:** COMUNE DI MONTESCUDAO (c_f640, B3002110506, Comuni e loro Consorzi e Associazioni, Pubbliche Amministrazioni, www.comune.montescudao.pi.it).

At the bottom right of the search results, it says 'Presenti 15.319 occorrenze. Mostrate dalla 1 alla 12.'

Figure5.33: Menù «Esplora Sezioni» - Filtro «Prevenzione della corruzione» - tutte le Amministrazioni

TrasparenzAI - Piattaforma per l'analisi e la consultazione della trasparenza amministrativa delle Pubbliche Amministrazioni, Release 1.0.0

Protocollo d'intesa tra l'Autorità Nazionale Anticorruzione e il Consiglio Nazionale delle Ricerche - 7 agosto 2023 | CREDITS | DOCUMENTAZIONE | ID: IVAN DUCA > | Cerca amministrazione |

TrasparenzAI
Piattaforma per l'analisi e la consultazione della trasparenza amministrativa delle Pubbliche Amministrazioni

Mappa delle Amministrazioni | Regole | Grafici e Mappe | Esplora Sezioni

Filtra la tua ricerca

Controllo
Controllo del 07/06/2025 19:00:00, stato COMPLETATO | Nome della Regola
Prevenzione della Corruzione | Regole figlie non trovate | Stato
Non Trovato

Denominazione Ente | Codice IPA | Categoria | Ordinamento | Scarica in CSV | Presenti 284 occorrenze. Mostrare dalla 1 alla 12.

LOREO MULTISERVIZI SRL Codice IPA: TLU1ZPKZ Codice Fiscale: 01385890429 Categoria: Gestori di Pubblici Servizi Tipologia: Gestori di Pubblici Servizi Sito istituzionale: WWW.LOREOTOMULTISERVIZI.IT Stato: Non Trovato Aggiornato al: 07/06/2025 19:03:07 Nome della Regola: prevenzione-corruzione Termine: Prevenzione della Corruzione, Piano triennale, Prevenzione corruzione ricercato: triennale, Prevenzione corruzione	ORDINE DELLA PROFESSIONE INTERPROVINCIALE DI PARMA E PIACENZA Codice IPA: cdo_034 Codice Fiscale: 80003230341 Categoria: Federazioni Nazionali, Ordini, Collegi e Consigli professionali Tipologia: Pubbliche Amministrazioni Sito istituzionale: www.oipoiprprc.it Stato: Non Trovato Aggiornato al: 07/06/2025 19:04:19 Nome della Regola: prevenzione-corruzione Termine: Prevenzione della Corruzione, Piano triennale, Prevenzione corruzione ricercato: triennale, Prevenzione corruzione	LO SCRIGNO MAGICO - SOCIETA' COOPERATIVA SOCIALE Codice IPA: Ismscs Codice Fiscale: 04812320960 Categoria: Gestori di Pubblici Servizi Tipologia: Gestori di Pubblici Servizi Sito istituzionale: www.scrignomagico.com Stato: Non Trovato Aggiornato al: 07/06/2025 19:08:53 Nome della Regola: prevenzione-corruzione Termine: Prevenzione della Corruzione, Piano triennale, Prevenzione corruzione ricercato: triennale, Prevenzione corruzione
ENTE NAZIONALE RISI Codice IPA: enr Codice Fiscale: 03036460156 Categoria: Stazioni Appaltanti Tipologia: Stazioni Appaltanti Sito istituzionale: www.entrarsi.it Stato: Non Trovato Aggiornato al: 07/06/2025 19:09:14 Nome della Regola: prevenzione-corruzione Termine: Prevenzione della Corruzione, Piano triennale, Prevenzione corruzione ricercato: triennale, Prevenzione corruzione	COLLEGIO GEOMETRI E GEOMETRI LAUREATI DI PADOVA Codice IPA: cggipd Codice Fiscale: 80017020282 Categoria: Federazioni Nazionali, Ordini, Collegi e Consigli professionali Tipologia: Pubbliche Amministrazioni Sito istituzionale: www.geometri.it Stato: Non Trovato Aggiornato al: 07/06/2025 19:09:19 Nome della Regola: prevenzione-corruzione Termine: Prevenzione della Corruzione, Piano triennale, Prevenzione corruzione ricercato: triennale, Prevenzione corruzione	POLITECNICO DELLE ARTI DI BERGAMO "G. DONIZETTI - G. CARRARA" Codice IPA: Ismg_016 Codice Fiscale: 03537600169 Categoria: Istituzioni per l'Alta Formazione Artistica, Musicale e Coreutica - AFAM Tipologia: Pubbliche Amministrazioni Sito istituzionale: www.coreutica.it Stato: Non Trovato Aggiornato al: 07/06/2025 19:11:21 Nome della Regola: prevenzione-corruzione Termine: Prevenzione della Corruzione, Piano triennale, Prevenzione corruzione ricercato: triennale, Prevenzione corruzione

| | |

Figure5.34: Menù «Esplora Sezioni» - Filtro «Prevenzione della corruzione» - stato «non trovato»

Protocollo d'intesa tra l'Autorità Nazionale Anticorruzione e il Consiglio Nazionale delle Ricerche - 7 agosto 2023 | CREDITS | DOCUMENTAZIONE | ID: IVAN DUCA > | Esci | Configurazione |

TrasparenzAI
Piattaforma per l'analisi e la consultazione della trasparenza amministrativa delle Pubbliche Amministrazioni

Mappa delle Amministrazioni | Regole | Grafici e Mappe | Esplora Sezioni

Presenza della sezione Amministrazione Trasparente in Homepage

Controllo del 07/06/2025	Analisi del 07/06/2025	Legenda												
COMPLETATO PA Analizzate: 23.674 Homepage conforme: 18.099 Homepage conforme con testo alternativo: 720 Homepage non conforme: 1.987 Sito PA non raggiungibile: 1.978 Sito PA non dichiarato: 990	<table border="1"> <caption>Data for the pie chart</caption> <thead> <tr> <th>Categoria</th> <th>Percentuale</th> </tr> </thead> <tbody> <tr> <td>Homepage conforme</td> <td>4,18%</td> </tr> <tr> <td>Homepage conforme con testo alternativo</td> <td>7,93%</td> </tr> <tr> <td>Homepage non conforme</td> <td>8,39%</td> </tr> <tr> <td>Sito PA non raggiungibile</td> <td>3,04%</td> </tr> <tr> <td>Sito PA non dichiarato</td> <td>8,39%</td> </tr> </tbody> </table>	Categoria	Percentuale	Homepage conforme	4,18%	Homepage conforme con testo alternativo	7,93%	Homepage non conforme	8,39%	Sito PA non raggiungibile	3,04%	Sito PA non dichiarato	8,39%	<ul style="list-style-type: none"> Homepage conforme Il sito contiene la sezione 'Amministrazione Trasparente' come previsto dal D.Lgs. 33/2013 Homepage conforme con testo alternativo La sezione 'Amministrazione Trasparente' è raggiungibile attraverso un link in homepage non rispondente al dettato del D.Lgs. 33/2013 Homepage non conforme Il sito non contiene la sezione 'Amministrazione Trasparente' come previsto dal D.Lgs. 33/2013 Sito PA non raggiungibile Il sito fornito dall'Amministrazione non risulta raggiungibile Sito PA non dichiarato
Categoria	Percentuale													
Homepage conforme	4,18%													
Homepage conforme con testo alternativo	7,93%													
Homepage non conforme	8,39%													
Sito PA non raggiungibile	3,04%													
Sito PA non dichiarato	8,39%													

Figure5.35: Menù «Accesso area riservata»

5.10. Menù «Accesso area riservata»

58

The screenshot shows the configuration interface of the TrasparenzAI platform. At the top, there's a navigation bar with links for 'CREDITS', 'DOCUMENTAZIONE', 'ID IVAN DUCA', and a search bar. Below the header, there are tabs for 'Mappa delle Amministrazioni', 'Regole', 'Grafici e Mappe', and 'Esplora Sezioni'. The main content area is divided into several sections:

- Impostazioni controllo:** Includes a URL field set to 'https://monitorai.ba.cnr.it/conductor-server/api/workflow' and a 'Conferma' button.
- PARAMETRI:** A section for rule parameters. It shows a dropdown for 'Nome della Regola' set to 'AT_TO_BE_23-12-2024 - Amministrazione Trasparente', and fields for 'Timeout di connessione iniziale (ms)' (30000), 'Timeout di risposta iniziale (ms)' (60000), 'Timeout di connessione massimo (ms)' (60000), and 'Timeout di risposta massimo (ms)' (120000). It also includes sections for 'HTTP STREAM' and 'START WORKFLOW' with their respective URLs.
- Impostazioni colori:** A section for color mapping. It lists six categories with corresponding hex codes: 'Conforme' (#ff4a047), 'Conforme con testo' (#66bb6a), 'alternativo' (#ff0300), 'Non conforme' (#fb8c00), 'Non raggiungibile' (#b0bec5), and 'Pagina non trovata' (#ffe000). Each entry has a 'Conferma' button.

The footer of the page contains the following elements:

- The TrasparenzAI logo and text: 'TrasparenzAI - Piattaforma per l'analisi e la consultazione della trasparenza amministrativa delle Pubbliche Amministrazioni'.
- A note about the project: 'Progetto per la sperimentazione di tecnologie digitali innovative per l'analisi automatica della sezione Amministrazione Trasparente dei siti web delle Pubbliche Amministrazioni e dei soggetti giuridici che rientrano nel campo di applicazione del D.Lgs. n. 33/2013.'
- Logos for 'Consiglio Nazionale delle Ricerche' and 'AUTORITÀ NAZIONALE ANTICORRUZIONE'.
- Links for 'Note legali' and 'Privacy policy'.

Figure5.36: Menù «Configurazione»

CHAPTER 6

Appendice

6.1 Autori

Autore del codice: Ivan Duca <ivan.duca@cnr.it>

Autore del codice: Dario Elia <dario.elia@cnr.it>

Autore del codice: Claudia Greco <claudia.greco@cnr.it>

Autore del codice: Massimo Ianigro <massimo.ianigro@cnr.it>

Autore del codice: Cristian Lucchesi <cristian.lucchesi@cnr.it>

Autore del codice: Marco Spasiano <marco.spasiano@cnr.it>

6.2 Glossario

Amministrazione Trasparente

Sezione obbligatoria dei siti delle Pubbliche Amministrazioni prevista dal D.Lgs. 33/2013, contenente dati e documenti per garantire trasparenza e accountability.

ANAC

Autorità Nazionale Anticorruzione, organismo di controllo e prevenzione della corruzione nella Pubblica Amministrazione. Partner del progetto TrasprenzAI.

CNR

Consiglio Nazionale delle Ricerche. Ente pubblico di ricerca nazionale con competenze multidisciplinari, vigilato dal Ministero dell'Università e della Ricerca (MUR).

IndicePA

Fonte OpenData ufficiale che elenca tutte le Pubbliche Amministrazioni italiane, utilizzata per alimentare la base dati della piattaforma.

Keycloak

Identity Provider opensource adottato per gestire l'autenticazione OAuth2 e la gestione dei ruoli utente.

Microservizi

Architettura software modulare in cui ogni funzione di TrasparenzAI è realizzata come servizio indipendente e integrato.

OpenAPI / Swagger UI

Strumenti per la documentazione automatica e l'interazione diretta con le API REST esposte dai microservizi.

Selenium Grid

Infrastruttura distribuita per il rendering delle pagine web dinamiche, basata su istanze multiple di browser Chrome.

Traefik

Reverse proxy e bilanciatore di carico che gestisce il traffico HTTPS verso i microservizi.

Indice

A

Amministrazione Trasparente, **60**
ANAC, **60**

C

CNR, **60**

I

IndicePA, **60**

K

Keycloak, **60**

M

Microservizi, **61**

O

OpenAPI / Swagger UI, **61**

S

Selenium Grid, **61**

T

Traefik, **61**