TOURISM DIGITAL HUB-TDH022

LINEE GUIDA SULL'INTEROPERABILITÀ TECNICA E LA GESTIONE DELLE API

Documento operativo

Raccomandazioni di Implementazione

Versione 0.1 del 15/12/2021

Versione	Data	Tipologia Modifica
0.1	15/12/2021	Prima Release

Indice Generale

CAPITOLO 1 – INTRODUZIONE

Il presente Documento Operativo riporta le indicazioni che gli erogatori *(in tal senso si considerano Soggetti Pubblici quali, a titolo esemplificativo Regioni e Province, oltre che Enti Pubblici o assimilabili e Soggetti Privati, incluse Seconde e Terze Parti che mettono a disposizione del TDH servizi e funzionalità)* attestati all'interno del Tourism Digital Hub devono considerare nell'implementazione delle API al fine di favorire l'interoperabilità con i fruitori, anch'essi attestati all'interno del Tourism Digital Hub *(in tal senso si considerano invece tutti i soggetti che utilizzano i servizi digitali messi a disposizione dagli erogatori all'interno dell'Ecosistema)*.

Le raccomandazioni sono applicate dagli erogatori in funzione alle specifiche esigenze applicative e/o in relazione alla natura dei fruitori.

Questo documento, la cui applicazione è relativa al contesto specifico del Tourism Digital Hub (TDH), è stato redatto sulla base del Documento Operativo "Raccomandazioni di Implementazione"[1] emanato da AgID e collegato al documento "Linee Guida sull'interoperabilità tecnica delle Pubbliche Amministrazioni"[2] sempre emanato da AgID; in tal senso si rimanda ai due documenti sopracitati per i principi generali.

[1] Riferimento online: https://www.agid.gov.it/sites/default/files/repository_files/04_raccomandazioni_di_implementazione.pdf
[2] Riferimento online: https://www.agid.gov.it/sites/default/files/repository_files/linee_guida_interoperabilit_tecnica_pa.pdf

CAPITOLO 2 – AMBITO DI APPLICAZIONE

Il presente Documento Operativo è redatto quale documento operativo relativo alla Linea di indirizzo sull'interoperabilità tecnica.

2.1 Soggetti destinatari del documento

Il Documento Operativo è destinato a tutti quei Soggetti (Pubblici e Privati, in precedenza definiti erogatori) che mettono a disposizione dei fruitori servizi e funzionalità all'interno del Tourism Digital Hub (TDH) oltre che agli stessi fruitori nelle more della fruizione dei servizi e delle funzionalità desiderate; queste disposizioni possono dunque essere utilizzare come base per implementazione di nuove funzionalità nel caso in cui debbano essere sviluppate ex-novo ovvero come base per integrazione delle funzionalità esistenti.

Di seguito, *a livello esemplificativo e non esaustivo*, si riporta un elenco dei Soggetti Pubblici e Privati destinatari del Documento Operativo, sia presenti a titolo di erogatori che di fruitori dei servizi e delle funzionalità all'interno del Tourism Digital Hub (TDH).

Soggetti Pubblici

- Pubblica Amministrazione Centrale (es. Ministero del Turismo),
- Pubblica Amministrazione Locale (es. Regioni, Province…),
- Enti Nazionali e Locali (es. ENIT),
- Enti No Profit,
- Imprese pubbliche collegate agli ambiti turistici (es. impianti di risalita…).

Soggetti Privati

- Imprese ricettive, di ristorazione, ecc…,
- Tour Operator/Agenzie di viaggio,
- Sindacati,
- Imprese private collegate agli ambiti turistici (es. impianti di risalita…).

CAPITOLO 3 – RIFERIMENTI E SIGLE

3.1 Note di lettura del documento

Conformemente alle norme ISO/IEC Directives, Part 3 per la stesura dei documenti tecnici il presente Documento Operativo utilizzerà le parole chiave «DEVE», «DEVONO», «NON DEVE», «NON DEVONO», «DOVREBBE», «NON DOVREBBE», «PUÒ» e «OPZIONALE», la cui interpretazione è descritta di seguito:

- **DEVE o DEVONO**, indicano un requisito obbligatorio per rispettare la Linea di indirizzo;
- **NON DEVE o NON DEVONO**, indicano un assoluto divieto delle specifiche;
- **DOVREBBE o NON DOVREBBE**, indicano che le implicazioni devono essere comprese e attentamente pesate prima di scegliere approcci alternativi;
- **PUÓ o POSSONO o l'aggettivo OPZIONALE**, indica che il lettore può scegliere di applicare o meno senza alcun tipo di implicazione o restrizione la specifica

3.2 Termini e definizioni[3]

Per una più agevole lettura si riporta un glossario dei termini e delle definizioni contenuti nel presente documento.

[AgID]	Agenzia per l'Italia Digitale
[API]	Application Programming Interface
[Array]	Struttura dati complessa, statica ed eterogenea
[CAD]	Decreto Legislativo 7 marzo 2005, n. 82 - «Codice dell'Amministrazione Digitale» (noto anche come "CAD"), aggiornato con modifiche dal D.L. 16 luglio 2020 n.76 e convertito in legge con la L. 11 settembre 2020 n.120
[E2E-Key]	Chiave univoca della transazione

[3] Alcuni termini e definizioni esplicitati all'interno di questo paragrafo sono presenti anche all'interno del documento di "Linee Guida sull'interoperabilità tecnica delle Pubbliche Amministrazioni" emanate da AgID (si rimanda alla sezione "Bibliografia e Sitografia di Riferimento" per i link di redirect ai contenuti citati)

[Erogatore]	Uno dei soggetti di cui all'articolo 2, comma 2 del CAD che rende disponibile e-service ad altre organizzazioni, per la fruizione di dati in suo possesso o l'integrazione dei processi da esso realizzati
[Fruitore]	Organizzazione che utilizza gli e-service messi a disposizione da un dei soggetti di cui all'articolo 2, comma 2 del CAD
[HTTP]	Hypertext Transfer Protocol
[IDL]	Interface Description Language
[JSON]	JavaScript Object Notation
[Open API]	Specifica per gestire servizi web RESTful
[REST]	Representational State Transfer
[Req-Timestamp]	Indicazione del momento in cui è stata effettuata una richiesta
[SOAP]	Simple Object Access Protocol
[TDH]	Tourism Digital Hub
[TDH022]	TDH022 - Interfaccia di interoperabilità del Tourism Digital Hub
[UTF-8]	Unicode Transformation Format, 8 bit
[WSDL]	Web Services Description Language

CAPITOLO 4 – RACCOMANDAZIONI TECNICHE DI IMPLEMENTAZIONE

Il contenuto di questo capitolo contiene le raccomandazioni tecniche che gli erogatori devono tenere presente ai fini della realizzazione delle API da esporre all'interno del Tourism Digital Hub (TDH).

[RAC_GEN_001] Descrizione delle API

Le API DEVONO essere rappresentate mediante un Interface Description Language standard (IDL). Nello specifico:

- **REST:** Swagger 2.0, RAML 1.0, OpenAPI 3.0 e successive;
- **SOAP:** WSDL 1.1 e successive.

[RAC_GEN_002] Endpoint delle API

Il numero di versione NON DEVE essere presente all'interno del nome della API.

Si DOVREBBE indicare il numero di versione e la tecnologia nell'endpoint delle API.

Esempio:

```
http://<dominioOrganizzativo>-<NomeAPI>-
[rest|soap]/<DominioApplicativo>/v<major>[.<minor>[.<patch>]]/<risorsa>
```

dove:

- **<dominioOrganizzativo>** indica l'organizzazione che espone il servizio;
- **[rest|soap]** indica la tecnologia della API;
- **<DominioApplicativo>** indica il settore all'interno dell'organizzazione;
- **v<major>[.<minor>[.<patch>]]** indica il numero di versione in coerenza con Semantic Versioning 2.0.01;
- **<NomeAPI>** è il nome della specifica API;
- **<risorsa>** indica il percorso logico (path), anche composto, per accedere alla risorsa a cui si fa riferimento.

[RAC_GEN_003] Codifica di default

Si DOVREBBE utilizzare UTF-8 come codifica di default per i dati.

[RAC_GEN_005]

Nel Soap Header o nell'Header del http (in caso di chiamata REST) andranno inviati i seguenti dati minimi:

- **Source:** indica in maniera univoca il sorgente del messaggio. Tale campo è di tipo stringa, e sarà così composto: *<ragionesociale>_<progressivo>* dove:
  - *<ragionesociale>* indica la ragione sociale del soggetto chiamante,
  - *<progressivo>* id univoco condiviso tra il soggetto chiamante e il TDH022.
- **Req-Timestamp:** indica il momento in cui è stata fatta la richiesta. Tale campo è di tipo data secondo quanto riportato in seguito nel [RAC_GEN_FORMAT_003];
- **e2e-Key:** rappresenta la chiave univoca della transazione. Il campo è di tipo stringa e avrà una lunghezza massima di 20 caratteri.

[RAC_REST_FORMAT_003] Convenzioni di rappresentazione

DEVONO usarsi le seguenti convenzioni di rappresentazione:

- I booleani non DEVONO essere null;
- Gli array vuoti non DEVONO essere null, ma liste vuote, ad es. [];
- Le enumeration DEVONO essere rappresentate da stringhe non nulle;
- I tipi "Data" non DEVONO essere null o vuoti.

[RAC_REST_NAME_007] Usare lo schema Problem JSON per le risposte di errore

Nel caso di errori gestiti dall' applicazione si DEVONO ritornare:

- Un payload di tipo Problem definito in RFC 7807;
- il media type application/problem+json;
- uno status code esplicativo;
- l'oggetto, possibilmente esteso.

Quando si restituisce un errore è importante non esporre dati interni delle applicazioni. Per prevenire il rischio di user-enumeration, i messaggi di errore di autenticazione non devono fornire informazioni sull'esistenza o meno dell'utenza.

Dopo aver validato il contenuto delle richieste si DEVE ritornare un codice http di errore secondo la RFC 2616.

CAPITOLO 5 – ROBUSTEZZA[4]

Ai fini di garantire la responsività di una API è necessario impedire a singoli fruitori di esaurire la capacità di calcolo e di banda dell'erogatore. La tecnica comunemente utilizzata in questi casi è il rate limiting (anche noto come throttling). Il rate limit fornisce ad uno specifico fruitore un numero massimo di richieste soddisfacibili all'interno di uno specifico arco temporale (es. 1000 richieste al minuto). Un numero di richieste che superi il limite imposto provoca il rifiuto di ulteriori richieste da parte di uno specifico fruitore per un intervallo di tempo predeterminato.

Sulle politiche riguardanti il numero massimo di richieste e la relativa finestra temporale, e quelle riguardanti il tempo di attesa per nuove richieste (che può essere incrementato in caso di richieste reiterate, es. con una politica di aumento esponenziale) si lascia libertà agli implementatori previa un'analisi di carico massimo sopportabile dall'erogatore.

[RAC_ROBUSTEZZA_001] Segnalare raggiunti limiti di utilizzo

Gli erogatori di interfacce di servizio REST DEVONO segnalare eventuali limiti raggiunti con HTTP status 429 Too Many Requests.

Le API restituiscono in ogni risposta i valori globali di throttling tramite i seguenti header:

- **X-RateLimit-Limit:** limite massimo di richieste per un endpoint;
- **X-RateLimit-Remaining:** numero di richieste rimanenti fino al prossimo reset;
- **X-RateLimit-Reset:** numero di secondi che mancano al prossimo reset.

In caso di superamento delle quote, le API DOVREBBERO restituire anche l'header:

- **HTTP header Retry-After:** numero minimo di secondi dopo cui il client è invitato a riprovare.

Nel caso di SOAP non esistono regole guida standard per la gestione del rate limit e del throttling. Si POSSONO utilizzare gli stessi header e status code HTTP visti nel caso REST.

I fruitori DEVONO:

- rispettare gli header di throttling;

[4] Il contenuto dell'introduzione di questo paragrafo è in linea con quanto prescritto dal Documento Operativo "Raccomandazioni di Implementazione" collegato alle "Linee Guida sull'interoperabilità tecnica delle Pubbliche Amministrazioni" edite da AgID, di cui al capitolo 8 del Documento Operativo citato (si rimanda alla sezione "Bibliografia e Sitografia di Riferimento" per i link di redirect ai contenuti).

- rispettare l'header X-RateLimit-Reset quando restituisce il numero di secondi che mancano al prossimo reset, ed eventualmente gestire l'indicazione in timestamp unix;
- rispettare l'header HTTP header Retry-After sia nella variante che espone il numero di secondi dopo cui riprovare, sia nella variante che espone la data in cui riprovare.

[RAC_ROBUSTEZZA_002] Segnalare il sovraccarico del sistema o l'indisponibilità del servizio

Gli erogatori DEVONO definire ed esporre un piano di continuità operativa segnalando il sovraccarico del sistema o l'indisponibilità del servizio con HTTP status 503 Service Unavailable.

In caso di sovraccarico o indisponibilità, l'erogatore DOVREBBE ritornare anche:

- HTTP header Retry-After con il numero minimo di secondi dopo cui il client è invitato a riprovare.

I fruitori DEVONO:

- rispettare HTTP header Retry-After sia nella variante che espone il numero di secondi dopo cui riprovare, sia nella variante che espone la data in cui riprovare.

La gestione del Rate limit dovrebbe essere esterna al descrittore delle interfacce, attraverso una componente di API Management. Qualora il soggetto dovesse essere sprovvisto di tale componente DEVE prevedere nella descrizione delle API l'indicazione degli header relativi al rate limiting. L'utilizzo degli header HTTP in SOAP è fuori dagli obiettivi di WSDL come Interface Definition Language.

BIBLIOGRAFIA E SITOGRAFIA DI RIFERIMENTO

Linee Guida sull'interoperabilità tecnica delle Pubbliche Amministrazioni

Autore: AgID – Prima pubblicazione: 27/04/2021

Riferimento online:
https://www.agid.gov.it/sites/default/files/repository_files/linee_guida_interoperabilit_tecnica_pa.pdf

Documento Operativo – Raccomandazioni di Implementazione

Autore: AgID – Prima pubblicazione: 27/04/2021

Riferimento online:

https://www.agid.gov.it/sites/default/files/repository_files/04_raccomandazioni_di_implementazione.pdf

Immagine di copertina – Credits

Astratto vettore creata da vectorjuice - it.freepik.com