Protocollo di Comunicazione

Property Management System con

Sistema Informativo

Osservatorio Turistico Regione Siciliana

Metodi, analisi
e valutazioni economiche

SIMULATION
INTELLIGENCE

Project Manager Advice Società Cooperativa	Dott. Salvatore Leonardi	
Referente di Progetto Izi SpA	Dott. Andrea Modica	
Dirigente Osservatorio Turistico Regione Siciliana	Dott.ssa Annamaria La Vecchia	

Sommario

SOMMARIO **1**

1. INTRODUZIONE **4**

1.1 Documenti di riferimento 4

1.2 Scopo del Documento 4

1.3 Campo di applicazione 4

1.4 Note sul copyright 4

1.5 Acronimi, Termini e Definizioni 4

1.6 Revisioni del documento 5

2. IL SISTEMA INFORMATIVO **6**

2.1 Descrizione Generale 6

3. ARCHITETTURA DEL SISTEMA **7**

3.1 Configurazione Architettura di rete 7

4. PROTOCOLLO DI COMUNICAZIONE **9**

4.1 Comunicazione Property Management System (PMS) 9

4.1.1 Gestione Autenticazione 9

4.1.1.1 Login 9

4.1.1.2 Logout 10

4.1.2 Gestione Arrivi 10

4.1.2.1 Stay 10

4.1.2.2 Guest 12

4.1.2.3 Room 13

4.1.2.4 Esempio 14

4.1.3 Aggiornamento soggiorno 14

4.1.3.1 Esempio 14

4.1.4 Check-out 15

4.1.4.1 Esempio 15

4.1.5 Occupazione Camere 16

4.1.5.1 Esempio 16

4.1.6 Contenuto Della Risposta 17

4.1.7 Chiusura Giornaliera 21

4.1.7.1 Esempio 21

4.2 File XSD corpo messaggi protocollo di comunicazione 22

4.2.1 Gestione Soggiorni 22

4.2.2 Fine Giornata 23

5. INIZIALIZZAZIONE SISTEMA 24

6. ESEMPI C# 25

6.1 Login 25

6.2 Nuovo Soggiorno / Modifica / Chiusura Giornata 26

1. Introduzione

1.1 Documenti di riferimento

Titolo	Autore	Data
Progetto per l'utilizzo dei fondi derivanti dalla sponsorizzazione (art. 4 del capitolato Speciale d'Appalto)	Regione Siciliana Assessorato del turismo, delle comunicazioni e dei trasporti. Dipartimento turismo, sport e spettacolo	Dicembre 2013
Progettazione Esecutiva	Advice Società Cooperativa	Giugno 2014

1.2 Scopo del Documento

Scopo del presente documento è quello di fornire le informazioni necessarie per consentire alle società produttrici di Property Management System di poter comunicare con il Sistema Informativo dell'Osservatorio Turistico della Regione Siciliana.

1.3 Campo di applicazione

Questo documento rappresenta uno degli elaborati tecnici relativi alla fase di messa in esercizio del Sistema Web dell'Osservatorio Turistico Regione Siciliana. Il documento raccoglie tutte le informazioni per poter adempiere agli obblighi previsti dalla legge e dal decreto assessoriale.

Al suo interno viene fornita tutte le informazioni relative ai metodi da invocare per implementare la comunicazione da parte delle strutture ricettive che possiedono un software gestionale.

1.4 Note sul copyright

Il presente documento ed i suoi contenuti sono di proprietà della Regione Siciliana e sono protetti dalle norme sul diritto d'autore e dalle altre norme applicabili. Il presente documento ed i suoi contenuti sono messi a disposizione sulla base dei termini definiti nei paragrafi precedenti.

1.5 Acronimi, Termini e Definizioni

Acronimo	Descrizione Estesa
API	Application Programming Interface
DB	Data Base
HTTPS	Hyper Text Transfer Protocol Secure
PA	Pubbliche Amministrazioni
PMS	Property Management System

SOA	Service Oriented Architecture
SOAP	Simple Object Access Protocol
SOA	Service Oriented Architecture
URL	Uniform Resource Locator
WSDL	Web Services Description Language
XML	eXtesible Markup Language

1.6 Revisioni del documento

Revisioni del documento

Data	Versione #	Autore	Descrizione dei cambiamenti
09/08/2014	1.0.0	Salvo Leonardi	Creazione documento
04/09/2014	1.0.1	Salvo Leonardi	Modificato il tipo Address
05/09/2014	1.0.2	Salvo Leonardi	Modificato nomi dei metodi
09/09/2014	1.0.3	Salvo Leonardi	Aggiunti file xsd
15/12/2014	1.0.4	Salvo Leonardi	Esempi C#
23/12/2014	1.0.5	Salvo Leonardi	Procedura di inizializzazione
20/03/2015	1.0.6	Salvo Leonardi	Modificati XML di esempio
03/04/2025	1.0.7	Salvo Leonardi	Aggiornamento endpoint

2. Il Sistema Informativo

2.1 Descrizione Generale

Il Sistema Informativo dell'Osservatorio Turistico della Regione Siciliana nasce dall'esigenza di gestire e monitorare le informazioni statistiche sui flussi turistici.

In particolare, l'Osservatorio Turistico regionale ha manifestato una certa criticità nell'acquisizione dei dati statistici sugli arrivi e presenze turistiche, il cui flusso, dagli operatori alle province e da queste all'Osservatorio, è molto lento.

Da questo punto di vista il fabbisogno dell'Osservatorio è quello di garantire alla totalità delle strutture ricettive regionali la possibilità di disporre di un strumento informatico di semplice utilizzo per l'invio dei dati obbligatori di movimentazione dei turisti per la redazione delle statistiche Istat ufficiali sul turismo.

A tale scopo di è ritenuto indispensabile la predisposizione di una piattaforma informatica di facile utilizzo in modo da rendere più fluido il processo.

Gli obiettivi di questo sistema sono molteplici e tendono alla creazione di una banca dati unica e completa di tutte le attività turistiche, consentendo di avere un elenco completo e sempre aggiornato di dati relativamente a:

- Strutture ricettive;
- Movimento dei clienti nelle strutture ricettive; • Disponibilità e occupazione delle camere;
- Dati geo spaziali delle strutture ricettive.

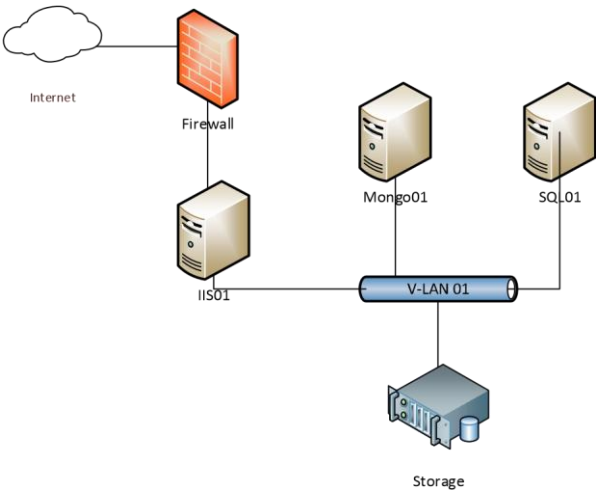
Inoltre il progetto si prefigge di:

- Agevolare l'operatore turistico nell'eseguire i propri adempimenti;
- Agevolare il procedimento amministrativo delle P.A. coinvolte e lo scambio dei dati;
- Agevolare tutti coloro che possono trarre beneficio dall'ottenimento delle informazioni.

3. Architettura del Sistema

L'architettura del sistema è basata su un approccio Web based, in modo da consentire l'utilizzo della soluzione indipendentemente dalla piattaforma utilizzata (Windows, Linux, Mac OS).

3.1 Configurazione Architettura di rete



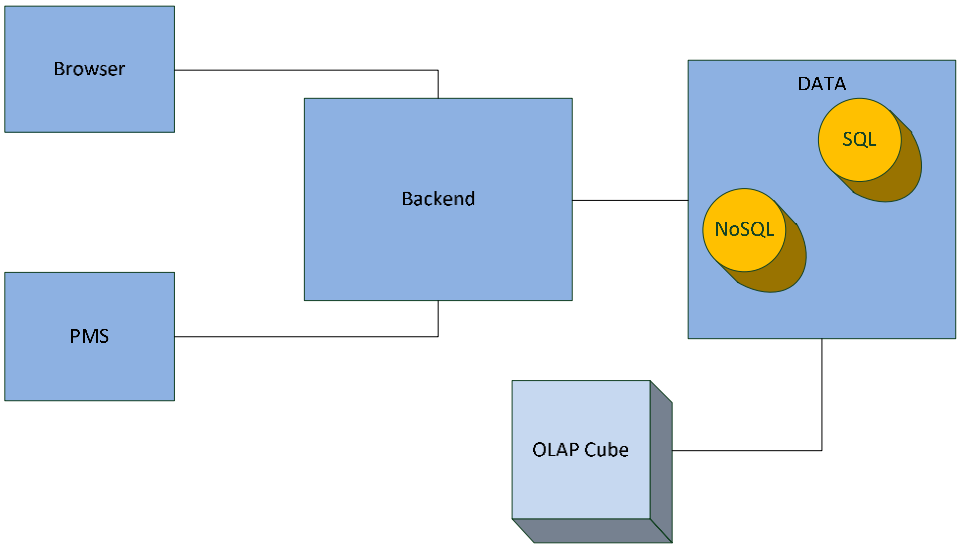


Figura 1 Architettura del Sistema

- **Browser**: Da interfaccia Web è possibile utilizzare il sistema tramite una serie di utilità che permettono facilmente di caricare le informazioni necessarie.
- **PMS**: E' possibile fornire i dati mediante PMS esistenti. Tali sistemi potranno caricare i dati necessari seguendo uno specifico protocollo (descritto in seguito)

- **Backend**: il cuore del sistema risiede nel backend. Questo componente riceverà le informazioni dall'esterno e, dopo una fase di validazione, le salverà a sistema. Inoltre permette ai browser di avere un esteso set di utilità per il caricamento e le analisi dei dati.
- **Data**: In questa sezione verranno memorizzati i dati, principalmente sono state utilizzati due tipi di sorgenti dati:
  - SQL: Per quanto riguarda i dati relativi alle anagrafiche delle strutture, profilazione, etc.
  - NoSQL: Per memorizzare i dati relativi ai flussi generati dai soggiorni.
- **OLAP Cube**: Sui dati raccolti può essere fatta un'analisi dettagliata grazie alla tecnologia OLAP e strumenti per la Business Intelligence.

Di seguito viene riportato uno schema di deployment:

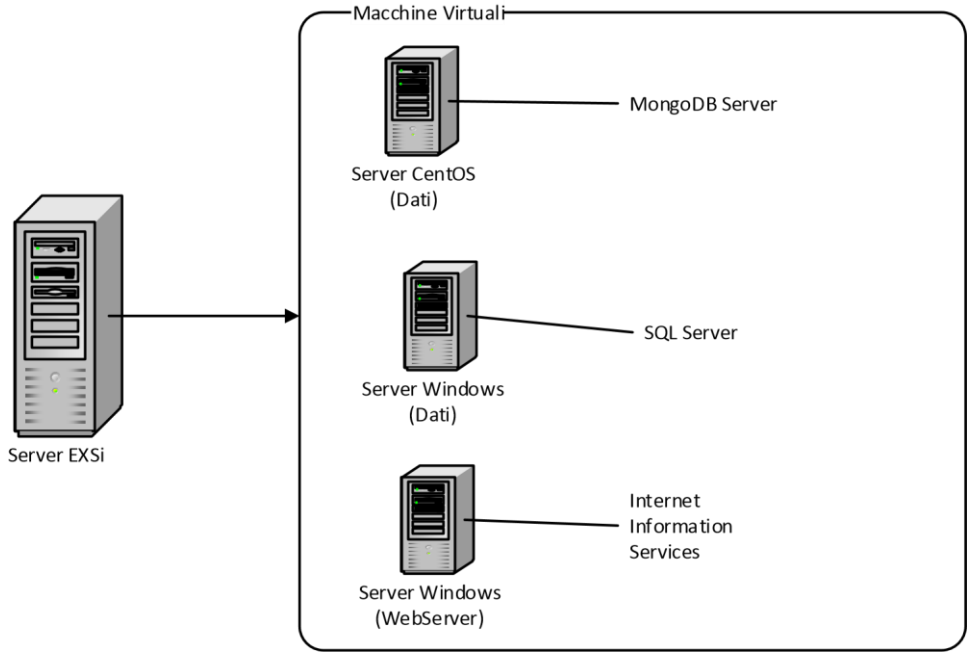


Figura 2 Schema di Deployment

4. Protocollo di comunicazione

4.1 Comunicazione Property Management System (PMS)

Le strutture che utilizzano un PMS possono inviare i dati direttamente al sistema utilizzando le WebAPI messe a disposizione. Di seguito sono elencati i messaggi da inviare.

La comunicazione può avvenire real-time, cioè invocando l'opportuno metodo a seconda dell'operazione che si sta svolgendo nella struttura. Al fine di poter garantire che tutti gli adempimenti previsti vengono soddisfatti devono essere tracciate le seguenti operazioni:

- Check-in;
- Check-out;
- Cambio Camera;
- Fine Giornata

Tutti i comandi sono sotto autenticazione, il sistema utilizza OAUTH2, ma sono messi a disposizione delle softwarehouse due webAPI aggiuntive che di fatto nascondo l'autenticazione OAUTH2. Metodi di Servizio:

- Log-in;
- Log-out.

Pertanto prima di poter eseguire qualunque operazione è necessario recuperare un token valido utilizzando il metodo di login, a fine trasmissione è opportuno invocare il metodo di logout.

N.B. Il protocollo è Case Sensitive, per cui è necessario rispettare le lettere maiuscole e minuscole come indicato nelle descrizioni dei campi.

4.1.1 Gestione Autenticazione

I parametri verranno forniti al momento della sottoscrizione al sistema e saranno validi per una singola struttura ricettiva. Con le suddette credenziali non sarà possibile accedere al sistema via web, ma queste saranno abilitate solo alla trasmissione tramite API.

4.1.1.1 Login

Web-API URL: https://osservatorioturistico.regione.sicilia.it/webapi/api/auth/login
Web-API VERB: GET

Nell'header devono essere valorizzati i seguenti campi:

- UserId
- Password

In risposta si otterrà il token autorizzativo, tale token è presente sia nell'header che nel body:

In Particolare nell'header sarà presente il seguente campo:

Authorization:Bearer
eyJ0eXAiOiJKVbhjsadgjhhdakshkas1NiJ9.eyJpc3MiOiJodHRwOi8vYXV0aC5yZWdpb25lLnNpY2lsaWEudHVyaXN0YXQvdHJ1c3QiLCJhdWQiOiJodHR2ljaWxpYS50dXJpc3RhdC9hcHAiLCJuYmYiOjE0MDk1NTg3MDMsImV4cCI6

MTQwOTU5NDcwMywibmFtZWlkIjoibGJvcmRpZXJpIiwidW5pcXVlX25hbWUiOiJsYm9yZGllcmkiLCJhdXRobWV0aG9kIjoiaHR0cDovL3NjaGVtYXMubWljcm9zb2Z0LmNvbS93cy8yMDA4LzA2L2lkZW50aXR5L2F1dGhlbnRpY2F0aW9ubWV0aG9kL3Bhc3N3b3JkIiwiYXV0aF90aW1lIjoiMjAxNC0wOS0wMVQwODowNTowMi4zMDRaIiwiZW1haWwiOiJsdWNpYW5vLmJvcmRpZXJpQHJlZ2lvbmUuc2ljaWxpYS5pdCIsImh0dHA6Ly9pZGVudGl0eXNlcnZlci50aGlua3RlY3R1cmUuY29tL2NsYWltcy9wcm9maWxlY2xhaW1zL3R1cmlzdGF0cHJvZmlsZXMiOiJ7XCIyMDEzXCI

Se si vuole ottenere il token dal body della response ricordarsi di eliminare il carattere “ posto all’inizio e alla fine.

Il Token recuperato dovrà essere inserito nell’header nel campo “Authorization” per ogni altra chiamata al sistema.

4.1.1.2 Logout

Web-API URL: https://osservatorioturistico.regione.sicilia.it/webapi/api/auth/logout
Web-API VERB: POST

Nell’header devono essere valorizzati i seguenti campi:

- UserId
- Authorization

4.1.2 Gestione Arrivi

Web-API URL: https://osservatorioturistico.regione.sicilia.it/webapi/api/stay/addfrompms
Web-API VERB: POST

Tutti i campi del messaggio sono obbligatori fatta eccezione per il campo Email dell’ospite.

Questo messaggio va inviato all’inserimento di un nuovo soggiorno, è possibile inviare una lista di nuovi soggiorni in un unico messaggio, il contenuto di tale messaggio dovrà essere un xml strutturato nel modo seguente:

4.1.2.1 Stay

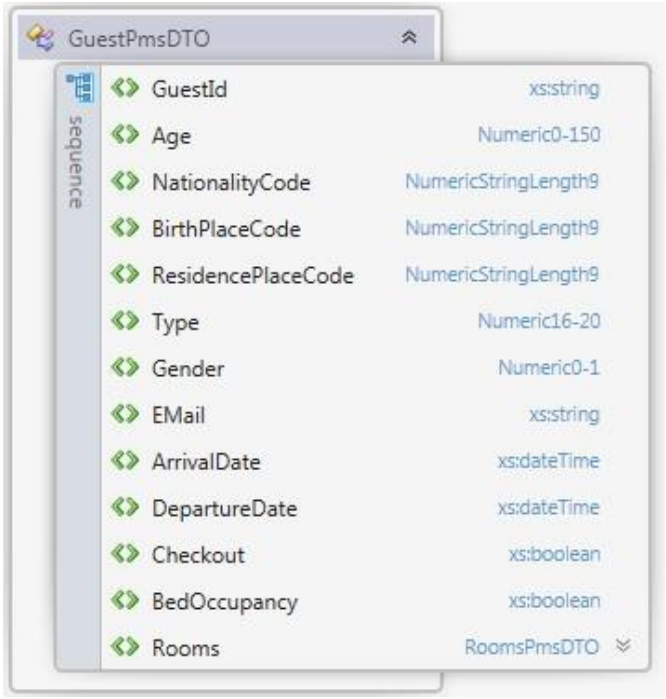
Nome Campo	Descrizione
StayId	Identificativo univoco del soggiorno
HotelCode	Codice Hotel rilasciato dal sistema in fase di registrazione
Guests	Lista degli ospiti coinvolti nel soggiorno


StaysPmsDTO
Stay [1..*]
StayPmsDTO
sequence
StayPmsDTO
sequence
HotelCode
xs:string
StayId
xs:string
Guests
Guest [1..*]
GuestPmsDTO

4.1.2.2 Guest

Questa sezione serve a inserire le informazioni di un singolo ospite al fine statistico

Nome Campo	Descrizione
GuestId	Codice che identifica univocamente un ospite
Age	Età dell'ospite
NationalityCode	Codice numerico (lunghezza 9) della tabella Nazioni del sistema AlloggiatiWeb della Polizia di Stato
BirthPlaceCode	Codice numerico (lunghezza 9) delle tabelle Nazioni e Comuni del sistema AlloggiatiWeb della Polizia di Stato. Deve essere passato il codice del Comune se Italiano, la nazione se Straniero
ResidencePlaceCode	Codice numerico (lunghezza 9) delle tabelle Nazioni e Comuni del sistema AlloggiatiWeb della Polizia di Stato. Deve essere passato il codice del Comune se Italiano, la nazione se Straniero
Type	Codice numerico (range 16-20) della definizione di tipologia di ospite nel sistema AlloggiatiWeb della Polizia di Stato.
Gender	Codice numerico (1, 2) della definizione di sesso nel sistema AlloggiatiWeb della Polizia di Stato.
EMail	Indirizzo email dell'ospite
ArrivalDate	Data di arrivo dell'ospite in struttura, utilizzare il formato UTC
DepartureDate	Data di uscita dell'ospite dalla struttura, utilizzare il formato UTC
Checkout	Boolean Flag che indica se l'ospite ha effettuato il checkout
BedOccupancy	Boolean Flag che indica se l'ospite occupa o meno un posto letto
Rooms	Lista delle stanze in cui l'ospite ha soggiornato durante la permanenza (più di una in caso di cambio camera)



4.1.2.3 Room

In questa sezione vanno inseriti le date di soggiorno per una determinata camera

Nome Campo	Descrizione
RoomId	Identificativo della camera
StartDate	Data di ingresso in camera, utilizzare il formato UTC
EndDate	Data di uscita dalla camera, utilizzare il formato UTC



Avere più camere per un singolo ospite all'interno di un soggiorno permette di gestire il cambio di camera da parte di un ospite all'interno dello stesso soggiorno al fine di poter recuperare l'informazione relativa all'occupazione netta, che deve essere giornalmente inviata all'ISTAT.

4.1.2.4 Esempio

Di seguito viene inserito un esempio di body:

```
<?xml version="1.0" encoding="utf-8" ?>
<StaysPmsDTO>
  <Stay>
    <StayId>TRS-IT-SIC-00004_2014_0714172522</StayId>
    <HotelCode>TRS-IT-SIC-00004</HotelCode>
    <Guests>
      <Guest>
        <GuestId>F4AD8F3AD7DF6F15EDB5E8F4BE83D4EB58051256</GuestId>
        <Age>25</Age>
        <NationalityCode>100000100</NationalityCode>
        <BirthPlaceCode>419087015</BirthPlaceCode>
        <ResidencePlaceCode>419082053</ResidencePlaceCode>
        <Type>16</Type>
        <Gender>1</Gender>
        <EMail>mario.r@mail.it</EMail>
        <ArrivalDate>2014-07-05T10:00:00.000Z</ArrivalDate>
        <DepartureDate>2014-07-11T10:00:00.000Z</DepartureDate>
        <Checkout>false</Checkout>
        <BedOccupancy>true</BedOccupancy>
        <Rooms>
          <Room>
            <RoomId>13</RoomId>
            <StartDate>2014-07-05T10:00:00.000Z</StartDate>
            <EndDate>2014-07-11T10:00:00.000Z</EndDate>
          <Room>
        <Rooms>
      </Guest>
    </Guests>
  </Stay>
</StaysPmsDTO>
```

4.1.3 Aggiornamento soggiorno

Web-API URL: https://osservatorioturistico.regione.sicilia.it/webapi/api/stay/updatefrompms
Web-API VERB: POST

Tutti i campi del messaggio sono obbligatori fatta eccezione per il campo Email dell'ospite.

Viene rinviato il soggiorno completo, con le variazioni apportate.

4.1.3.1 Esempio

Di seguito viene inserito un esempio di body:

```
<?xml version="1.0" encoding="utf-8" ?>
<StaysPmsDTO>
  <Stay>
```

```
    <StayId>TRS-IT-SIC-00004_2014_0714172522</StayId>
    <HotelCode>TRS-IT-SIC-00004</HotelCode>
    <Guests>
      <Guest>
        <GuestId>F4AD8F3AD7DF6F15EDB5E8F4BE83D4EB58051256</GuestId>
        <Age>25</Age>
        <NationalityCode>100000100</NationalityCode>
        <BirthPlaceCode>419087015</BirthPlaceCode>
        <ResidencePlaceCode>419082053</ResidencePlaceCode>
        <Type>16</Type>
        <Gender>1</Gender>
        <EMail>mario.r@mail.it</EMail>
        <ArrivalDate>2014-07-05T10:00:00.000Z</ArrivalDate>
        <DepartureDate>2014-07-13T10:00:00.000Z</DepartureDate>
        <Checkout>false</Checkout>
        <BedOccupancy>true</BedOccupancy>
        <Rooms>
          <Room>
            <RoomId>13</RoomId>
            <StartDate>2014-07-05T10:00:00.000Z</StartDate>
            <EndDate>2014-07-13T10:00:00.000Z</EndDate>
          <Room>
        <Rooms>
      </Guest>
    </Guests>
  </Stay>
</StaysPmsDTO>
```

4.1.4 Check-out

Web-API URL: https://osservatorioturistico.regione.sicilia.it/webapi/api/stay/updatefrompms
Web-API VERB: POST

Tutti i campi del messaggio sono obbligatori fatta eccezione per il campo Email dell'ospite.

Il Check-out serve ad indicare che uno o più ospiti hanno lasciato la struttura. All'interno del messaggio dovrà essere inviato lo stesso soggiorno indicato al check-in, con il campo "Checkout" dell'ospite che ha lasciato la struttura, settato a true.

4.1.4.1 Esempio

Di seguito viene inserito un esempio di body:

```
<?xml version="1.0" encoding="utf-8" ?>
<StaysPmsDTO>
  <Stay>
    <StayId>TRS-IT-SIC-00004_2014_0714172522</StayId>
    <HotelCode>TRS-IT-SIC-00004</HotelCode>
    <Guests>
      <Guest>
        <GuestId>F4AD8F3AD7DF6F15EDB5E8F4BE83D4EB58051256</GuestId>
        <Age>25</Age>
        <NationalityCode>100000100</NationalityCode>
        <BirthPlaceCode>419087015</BirthPlaceCode>
```

```
        <ResidencePlaceCode>419082053</ResidencePlaceCode>
        <Type>16</Type>
        <Gender>1</Gender>
        <Email>mario.r@mail.it</Email>
        <ArrivalDate>2014-07-05T10:00:00.000Z</ArrivalDate>
        <DepartureDate>2014-07-13T10:00:00.000Z</DepartureDate>
        <Checkout>true</Checkout>
        <BedOccupancy>true</BedOccupancy>
        <Rooms>
          <Room>
            <RoomId>13</RoomId>
            <StartDate>2014-07-05T10:00:00.000Z</StartDate>
            <EndDate>2014-07-13T10:00:00.000Z</EndDate>
          <Room>
        <Rooms>
      </Guest>
    </Guests>
  </Stay>
</StaysPmsDTO>
```

4.1.5 Occupazione Camere

Web-API URL: https://osservatorioturistico.regione.sicilia.it/webapi/api/stay/updatefrompms
Web-API VERB: POST

Tutti i campi del messaggio sono obbligatori fatta eccezione per il campo Email dell'ospite.

Qualora durante il soggiorno un ospite esegua un cambio camera, il sistema dovrà essere informato per aggiornare le informazioni relative all'occupazione delle camere da inviare all'ISTAT. Per notificare questo tipo di informazioni si dovrà inviare lo stesso soggiorno inviato al check-in ma con la modifica della sezione Room all'interno dell'ospite che ha effettuato il cambio.

4.1.5.1 Esempio

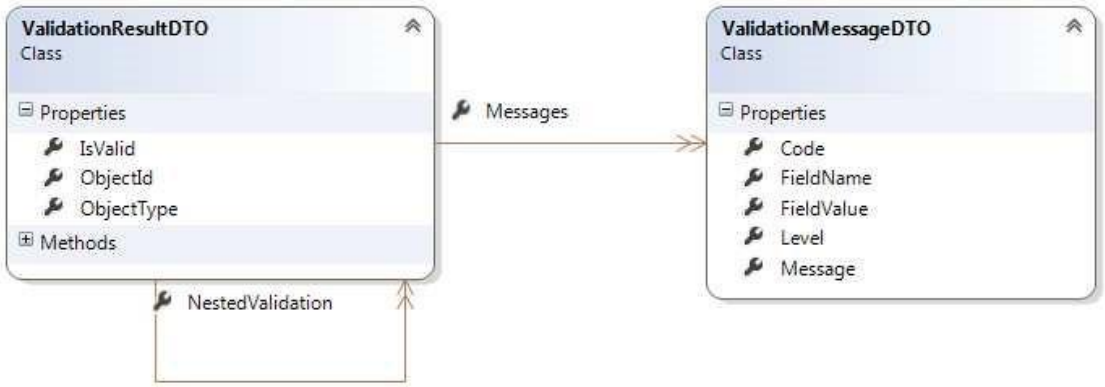
Di seguito viene inserito un esempio di body:

```
<?xml version="1.0" encoding="utf-8" ?>
<StaysPmsDTO>
  <Stay>
    <StayId>TRS-IT-SIC-00004_2014_0714172522</StayId>
    <HotelCode>TRS-IT-SIC-00004</HotelCode>
    <Guests>
      <Guest>
        <GuestId>F4AD8F3AD7DF6F15EDB5E8F4BE83D4EB58051256</GuestId>
        <Age>25</Age>
        <NationalityCode>100000100</NationalityCode>
        <BirthPlaceCode>419087015</BirthPlaceCode>
        <ResidencePlaceCode>419082053</ResidencePlaceCode>
        <Type>16</Type>
        <Gender>1</Gender>
        <EMail>mario.r@mail.it</EMail>
        <ArrivalDate>2014-07-05T10:00:00.000Z</ArrivalDate>
        <DepartureDate>2014-07-13T10:00:00.000Z</DepartureDate>
        <Checkout>false</Checkout>
        <BedOccupancy>true</BedOccupancy>
        <Rooms>
          <Room>
            <RoomId>13</RoomId>
            <StartDate>2014-07-05T10:00:00.000Z</StartDate>
            <EndDate>2014-07-11T10:00:00.000Z</EndDate>
          <Room>
<Room>
            <RoomId>14</RoomId>
            <StartDate>2014-07-12T10:00:00.000Z</StartDate>
            <EndDate>2014-07-13T10:00:00.000Z</EndDate>
          <Room>
        <Rooms>
      </Guest>
    </Guests>
  </Stay>
</StaysPmsDTO>
```

4.1.6 Contenuto Della Risposta

La risposta ai messaggi inviati, oltre il normale status code (200 se non vi sono problemi, 400 in caso di errore), contiene un xml che contiene dettagli aggiuntivi.

In particolare, tale xml, rappresenta la serializzazione della seguente struttura:



ValidationResultDTO:

Rappresenta il risultato della validazione relativa ad un oggetto

Nome Campo	Descrizione
IsValid	Se *true* l'oggetto di riferimento e tutti gli oggetti annidati sono validi, negli altri casi assume valore *false*
ObjectType	Tipo di oggetto a cui si riferisce
ObjectId	Identificativo dell'oggetto a cui si riferisce
Messages	Lista di oggetti *ValidationMessageDTO*, ogni elemento contiene informazioni riguardo l'errore verificatosi su un determinato campo dell'oggetto
NestedValidation	Lista di *ValidationResultDTO*, in cui ogni elemento rappresenta il risultato di validazione di uno degli oggetti annidati, se ve ne sono.

ValidationMessageDTO:

Contiene informazioni su un errore verificatosi per un campo dell'oggetto padre:

Nome Campo	Descrizione
Code	Codice di errore (usi futuri)
FieldName	Nome del campo che ha generato l'errore
FieldValue	Valore del campo che ha generato l'errore
Level	Livello di Errore (Error,Warning,Info)
Message	Messaggio descrittivo dell'errore generato

Esempio:
Si supponga di inviare il seguente soggiorno:

```
<?xml version="1.0" encoding="utf-8"?>
<StaysPmsDTO>
```

```
  <Stay>
    <HotelCode>123</HotelCode>
    <StayId>12345</StayId>
    <Guests>
      <Guest>
        <GuestId>mrss</GuestId>
        <Age>31</Age>
        <NationalityCode>100000100</NationalityCode>
        <BirthPlaceCode>419089017</BirthPlaceCode>
        <ResidencePlaceCode>419089017</ResidencePlaceCode>
        <Type>16</Type>
        <Gender>1</Gender>
        <EMail>m.r@</EMail>
        <ArrivalDate>2014-09-12T22:00:00Z</ArrivalDate>
        <DepartureDate>2014-09-14T22:00:00Z</DepartureDate>
        <Checkout>false</Checkout>
        <BedOccupancy>false</BedOccupancy>
        <Rooms>
          <Room>
            <RoomId>123</RoomId>
            <StartDate>2014-09-13T22:00:00Z</StartDate>
            <EndDate>2014-09-14T22:00:00Z</EndDate>
          </Room>
        </Rooms>
      </Guest>
    </Guests>
  </Stay>
</StaysPmsDTO>
```

Si noti che vi sono due errori:

1. Il formato della mail non è valido
2. Il soggiorno nella camera inizia il giorno dopo rispetto alla data di inizio soggiorno

Inviando tale messaggio otterremo, nel corpo della risposta (che avrà status code 400), il seguente xml:

```
<ArrayOfValidationResultDTO
xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="https://www.w3.org/2001/XMLSchema">
  <ValidationResultDTO>
    <ObjectType>Stay</ObjectType>
    <ObjectId>12345</ObjectId>
    <IsValid>false</IsValid>
    <Messages />
    <NestedValidation>
      <ValidationResultDTO>
        <ObjectType>StayGuest</ObjectType>
        <ObjectId>mrss</ObjectId>
        <IsValid>false</IsValid>
        <Messages>
          <ValidationMessageDTO>
            <Level>Error</Level>
            <Code>0</Code>
```

```
            <Message>Malformed email</Message>
            <FieldName>EMail</FieldName>
          </ValidationMessageDTO>
          <ValidationMessageDTO>
            <Level>Error</Level>
            <Code>0</Code>
            <Message>Arrival Date does not match Room Start
Date</Message>
            <FieldName>ArrivalDate</FieldName>
            <FieldValue>12/09/2014 22:00:00</FieldValue>
          </ValidationMessageDTO>
        </Messages>
        <NestedValidation />
      </ValidationResultDTO>
    </NestedValidation>
  </ValidationResultDTO>
</ArrayOfValidationResultDTO>
```

Tale messaggio sta indicando che il risultato per l'oggetto di tipo "Stay" con id "12345" ha avuto esito negativo. Inoltre sta indicando che il suo oggetto annidato "StayGuest" con id "mrss" non ha superato la validazione con due errori (infatti vi sono due elementi nella lista "Messages"):

Il primo contiene le seguenti informazioni:

```
<ValidationMessageDTO>
    <Level>Error</Level>
    <Code>0</Code>
    <Message>Malformed email</Message>
    <FieldName>EMail</FieldName>
</ValidationMessageDTO>
```

Ovvero che il campo "EMail" non ha un formato valido.

Il secondo invece:

```
 <ValidationMessageDTO>
    <Level>Error</Level>
    <Code>0</Code>
    <Message>Arrival Date does not match Room Start Date</Message>
    <FieldName>ArrivalDate</FieldName>
    <FieldValue>12/09/2014 22:00:00</FieldValue>
 </ValidationMessageDTO>
```

Indica che il campo "ArrivalDate" con valore "12/09/2014 22:00:00" va in conflitto con la data di inizio soggiorno (infatti essendoci una sola camera la data di inizio deve coincidere con la data di inizio soggiorno).

Correggendo il messaggio inviato otterremo la seguente riposta:

```
<ArrayOfValidationResultDTO
xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="https://www.w3.org/2001/XMLSchema">
  <ValidationResultDTO>
    <ObjectType>Stay</ObjectType>
    <ObjectId>12345</ObjectId>
```

```
    <IsValid>true</IsValid>
    <Messages />
    <NestedValidation>
      <ValidationResultDTO>
        <ObjectType>StayGuest</ObjectType>
        <ObjectId>mrss</ObjectId>
        <IsValid>true</IsValid>
        <Messages />
        <NestedValidation />
      </ValidationResultDTO>
    </NestedValidation>
  </ValidationResultDTO>
</ArrayOfValidationResultDTO>
```

In questo caso vediamo che sia l'oggetto "Stay" che i suoi oggetti annidati hanno avuto esito positivo.

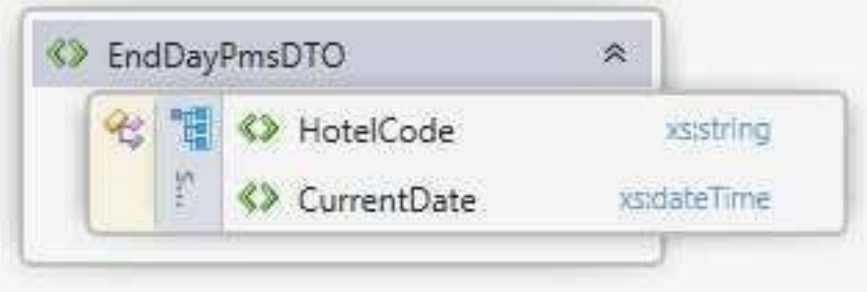
4.1.7 Chiusura Giornaliera

Web-API URL: https://osservatorioturistico.regione.sicilia.it/webapi/api/entity/enddayfrompms
Web-API VERB: POST

Tutti i campi del messaggio sono obbligatori.

Alla chiusura di giornata si dovrà mandare al sistema un messaggio che conterrà:

Nome Campo	Descrizione
CurrentDate	Data del giorno di cui si effettua la chiusura
HotelCode	Codice Hotel rilasciato dal sistema in fase di registrazione



Questo messaggio deve essere inviato sempre, in presenze e in assenza di movimenti di ospiti all'interno della struttura. Deve essere omesso durante le chiusure temporanee (ad esempio per le strutture stagionali).

4.1.7.1 Esempio

Di seguito viene inserito un esempio di body:

```
<?xml version="1.0" encoding="utf-8"?>
<EndDayPmsDTO>
  <HotelCode>TRS-IT-SIC-00004</HotelCode>
  <CurrentDate>2014-07-25T10:00:00.000Z</CurrentDate>
</EndDayPmsDTO>
```

4.2 File XSD corpo messaggi protocollo di comunicazione

4.2.1 Gestione Soggiorni

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema  xmlns:xs="https://www.w3.org/2001/XMLSchema">

  <xs:simpleType name="NumericStringLength9">
    <xs:restriction base="xs:string">
        <xs:pattern value="[0-9]{9}" />
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="Numeric16-20">
    <xs:restriction base="xs:unsignedInt">
      <xs:minInclusive value="16"/>
      <xs:maxInclusive value="20"/>
      <xs:pattern value="\d{1,2}"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="Numeric1-2">
    <xs:restriction base="xs:unsignedInt">
      <xs:minInclusive value="1"/>
      <xs:maxInclusive value="2"/>
      <xs:pattern value="\d{1}"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="Numeric0-150">
    <xs:restriction base="xs:unsignedInt">
      <xs:minInclusive value="0"/>
      <xs:maxInclusive value="150"/>
      <xs:pattern value="\d{1,3}"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name ="StayPmsDTO">
    <xs:sequence>
      <xs:element name="HotelCode" type="xs:string" />
      <xs:element name="StayId" type="xs:string" />
      <xs:element name="Guests">
        <xs:complexType>
          <xs:sequence>             <xs:element maxOccurs="unbounded"
name="Guest" type="GuestPmsDTO"/>           </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>

  <xs:complexType name = "RoomPmsDTO">
    <xs:sequence>
      <xs:element name="RoomId" type="xs:string" />
```

```
      <xs:element name="StartDate" type="xs:dateTime" />
      <xs:element name="EndDate" type="xs:dateTime" />
    </xs:sequence>
  </xs:complexType>


  <xs:complexType name="RoomsPmsDTO">
    <xs:sequence>
      <xs:element maxOccurs="unbounded" name="Room" type="RoomPmsDTO" />
    </xs:sequence>
  </xs:complexType>

  <xs:complexType name="GuestPmsDTO">
    <xs:sequence>
      <xs:element name="GuestId" type="xs:string"/>
      <xs:element name="Age" type="Numeric0-150" />
      <xs:element name="NationalityCode" type="NumericStringLength9" />
      <xs:element name="BirthPlaceCode" type="NumericStringLength9" />
      <xs:element name="ResidencePlaceCode" type="NumericStringLength9" />
      <xs:element name="Type" type="Numeric16-20" />
      <xs:element name="Gender" type="Numeric1-2" />
      <xs:element name="EMail" type="xs:string" />
      <xs:element name="ArrivalDate" type="xs:dateTime" />
      <xs:element name="DepartureDate" type="xs:dateTime" />
      <xs:element name="Checkout" type="xs:boolean" />
      <xs:element name="BedOccupancy" type="xs:boolean" />
      <xs:element name="Rooms" type="RoomsPmsDTO" />
    </xs:sequence>
  </xs:complexType>

  <xs:element name="StaysPmsDTO">
    <xs:complexType>
      <xs:sequence>
        <xs:element maxOccurs="unbounded" name="Stay" type="StayPmsDTO"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

4.2.2 Fine Giornata

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema xmlns:xs="https://www.w3.org/2001/XMLSchema">
  <xs:element name="EndDayPmsDTO">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="HotelCode" type="xs:string" />
        <xs:element name="CurrentDate" type="xs:dateTime" />
      </xs:sequence>
    </xs:complexType>
  </xs:element> </xs:schema>
```

5. Inizializzazione Sistema

Al momento dell'inizializzazione del sistema, quindi al primo invio dei dati, che decorrerà in data 01/01/2015, gli ospiti che già si trovano in struttura dovranno essere notificati al sistema tramite l'invio di un soggiorno fittizio che abbia data di inizio pari al 31/12/2014.

In questo modo il sistema terrà conto degli ospiti già presenti in struttura al momento del primo invio dati.

6. Esempi C#

In questo capitolo verranno forniti, a titolo di esempi, alcuni esempi in C#.

6.1 Login

Esempio in C# per ottenere il token autorizzativo dalla chiamata al metodo Login, tale token può essere prelevato sia dall'header che dal body della response.

Esempio di login con token prelevato dal body:

```
//Creazione richiesta
WebRequest request = WebRequest.Create("LoginUrl");
//Configurazione parametri
request.Method = "GET";
request.ContentType = "applcation/xml";
//Aggiunta delle credenziali nell'header
request.Headers.Add("UserId", username);
request.Headers.Add("Password", password);
//Chiamata al metodo login
var response = request.GetResponse();

//Recupero del token dal body della risposta
StreamReader reader = new StreamReader(response.GetResponseStream());
Token=  reader.ReadToEnd();

//Eliminazione dei caratteri di apertura e chiusura
Token=Token.Trim('"');

reader.Close();
reader.Dispose();
```

Esempio di login con token prelevato dall'header:

```
//Creazione richiesta
WebRequest request = WebRequest.Create("LoginUrl");
//Configurazione parametri
request.Method = "GET";
request.ContentType = "applcation/xml";
//Aggiunta delle credenziali nell'header
request.Headers.Add("UserId", username);
request.Headers.Add("Password", password);
//Chiamata al metodo login
var response = request.GetResponse();

//Recupero del token dall'header
String Token = response.Headers["Authorization"];
```

6.2 Nuovo Soggiorno / Modifica / Chiusura Giornata

Di Seguito viene riportato un esempio in C# per effettuare una delle operazioni di inserimento, modifica soggiorno e chiusura giornata. La differenza tra le tre operazioni sta nell'URL utilizzato per la creazione della richiesta e dal contenuto della stringa message (rappresentazione dell'xml da inviare)

```
//Creazione delle richiesta
HttpsWebRequest request=(HttpsWebRequest)WebRequest.Create("<Url>");
request.Method = "POST";
request.ContentType = "text/xml";
//Popolamento dell'Header con il token autorizzativo
request.Headers.Add("Authorization", Token);

//Inserimento del messaggio nel body della richiesta
Stream newStream = request.GetRequestStream();
byte[] data =
System.Text.Encoding.UTF8.GetBytes(message);
newStream.Write(data, 0, data.Length); newStream.Close();

//Lettura della response (ValidationResultDTO)
var response = request.GetResponse();
StreamReader reader = new StreamReader(response.GetResponseStream());
String res = reader.ReadToEnd();
reader.Close();
reader.Dispose();
```