

Tesina LWEB

Angelo Trifelli , Simone Amoriello

31 agosto 2022

Indice

1	Introduzione e descrizione dei dati	5
1.1	Tipi di utente	6
1.1.1	Visitatore	7
1.1.2	Cliente	7
1.1.3	Concierge	8
1.1.4	Amministratore	8
1.2	Casi d'uso	8
1.2.1	Visualizzazione informazioni relative all'hotel	9
1.2.2	Prenotazione di un soggiorno	9
1.2.3	Registrazione	10
1.2.4	Login	10
1.2.5	Visualizzazione e modifica delle informazioni personali	10
1.2.6	Pagamento mediante denaro e/o crediti	11
1.2.7	Servizio di ristorazione	11
1.2.8	Scrittura e commento di recensioni/commenti	12
1.2.9	Attività	13
1.2.10	Valutazione di recensioni e di commenti	14
1.2.11	Inserimento e risposta alle domande	14
1.2.12	Modifica del menù del ristorante	15
1.2.13	Modifica degli orari	15
1.2.14	Modifica delle attività	15
1.2.15	Visualizzazione delle prenotazioni dei clienti	16
1.2.16	Visualizzazione dei clienti attuali	16
1.2.17	Check-out di un cliente	16
1.3	Strutture dati	17
1.3.1	Clienti.xml	17
1.3.2	Amministratori.xml	20
1.3.3	Camere.xml	21
1.3.4	Ristorante.xml	23
1.3.5	Tavoli.xml	25

1.3.6	Attivita.xml	27
1.3.7	Recensioni.xml	30
1.4	Elementi XML	32

Capitolo 1

Introduzione e descrizione dei dati

Si propone di realizzare un sito web dedicato ad un hotel ed in grado di permettere all'utente di usufruire di tutte le varie funzionalità che l'hotel mette a disposizione.

Il sito è visitabile a partire da una pagina home iniziale, accessibile da tutte le tipologie di utenti, che descrive l'hotel fornendo informazioni su quest'ultimo, una serie di FAQ ed alcune attività che vengono messe a disposizione. Dalla home è poi possibile accedere ad una pagina dedicata per la prenotazione di una camera oppure direttamente alla schermata di login nel caso in cui l'utente è già cliente dell'hotel.

Il login (mediante username e password) può essere effettuato da tre tipologie di utenti:

- **Utente registrato:** cioè un utente che deve essere necessariamente registrato all'interno del sito dell'hotel. La registrazione può avvenire al momento del completamento della prima prenotazione di una camera oppure eventualmente l'utente potrà registrarsi anticipatamente (quindi senza dover necessariamente prenotare una camera).
- **Concierge:** utente moderatore che disporrà di credenziali personali e potrà svolgere operazioni volte alla moderazione del sito.
- **Admin:** un utente amministratore che disporrà di credenziali personali e che potrà effettuare operazioni riservate.

Nel caso in cui il login venga effettuato da un utente registrato allora si verrà portati in una pagina personale dedicata all'utente dalla quale, oltre

a poter ottenere un riepilogo dei propri dati personali (con la possibilità di modificarli) e del proprio soggiorno, sarà possibile scegliere di quali servizi si intende usufruire. In particolare viene messo a disposizione: un servizio di ristorazione ed un elenco di attività svolte all'interno dell'hotel.

Un cliente dell'hotel potrà inoltre scrivere una recensione che sarà visibile direttamente dalla home page del sito, ed avrà ad esso associata un proprio voto, una categoria ed eventuali commenti di risposta lasciati da altri utenti clienti dell'hotel. Per ogni recensione e per ogni commento è possibile inserire due ulteriori voti: il primo riguardante l'apprezzamento per il contenuto ed il secondo riguardante l'utilità della recensione o del commento (questi voti verranno inseriti da altri clienti dell'hotel).

Infine, un cliente dell'hotel avrà anche la possibilità di inserire una domanda riguardante una particolare categoria e ricevere risposte da altri clienti dell'hotel oppure dal concierge. Le domande con la risposta migliore potranno essere selezionate dalla concierge per essere elevate a FAQ.

Ogni cliente potrà inoltre accumulare dei crediti. Tali crediti verranno assegnati al momento del completamento di un soggiorno oppure periodicamente in base ad i giudizi ricevuti dalle proprie recensioni e commenti e potranno essere utilizzati per integrare un pagamento.

Se invece il login viene effettuato dal concierge o da un utente amministratore allora essi verranno portati in delle pagine personali, dalle quali potranno scegliere quale operazione intendono svolgere ed essere portati a pagine dedicate alla specifica operazione che si intende compiere.

1.1 Tipi di utente

Le tipologie di utenti in grado di utilizzare il sito sono le seguenti:

1. **Visitatore (utente non autenticato/non registrato)**
2. **Cliente (utente prenotato)**
3. **Concierge**
4. **Admin**

Vengono di seguito elencate le varie funzionalità associate a ciascuno di esso.

1.1.1 Visitatore

Un utente visitatore sarà in grado di:

1. Visualizzare le informazioni relative all'hotel reperibili dalla pagina home (comprese le FAQ e le recensioni).
2. Prenotare un soggiorno presso l'hotel.
3. Registrarsi presso il sito dell'hotel (nel caso in cui non sia già registrato)
4. Effettuare il login (nel caso in cui egli abbia completato la registrazione).

1.1.2 Cliente

Un utente che ha effettuato l'autenticazione mediante il login sarà in grado di:

1. Visualizzare e modificare le proprie informazioni personali.
2. Effettuare un pagamento mediante denaro e/o crediti.
3. Accedere alla home del servizio di ristorazione dalla quale potrà visualizzare il menù del ristorante, prenotare un tavolo o un servizio in camera e visualizzare/cancellare le eventuali prenotazioni già effettuate.
4. Accedere alla home delle attività offerte dall'hotel dalla quale potrà visualizzare le attività che vengono messe a disposizione, effettuare una prenotazione e visualizzare/cancellare le eventuali prenotazioni già effettuate.
5. Lasciare una recensione riguardante l'hotel oppure commentare una recensione o un commento già esistente.
6. Esprimere una valutazione su una recensione (o un commento) a cui è abilitato.
7. Inserire una domanda riguardante una particolare categoria oppure rispondere ad una domanda già esistente.

1.1.3 Conciierge

Un utente che si è autenticato come un utente moderatore (conciierge) sarà in grado di:

1. Modificare il menù del ristorante mediante l’inserimento di nuove portate oppure la modifica/cancellazione di portate già esistenti
2. Modificare gli orari del ristorante o di una specifica attività
3. Modificare le informazioni relative ad una specifica attività
4. Visualizzare tutte le prenotazioni di uno specifico cliente con la possibilità di modificare o annullare una specifica prenotazione
5. Elevare una particolare domanda effettuata da un cliente ed una specifica risposta a FAQ oppure inserire una nuova domanda come FAQ

1.1.4 Amministratore

Un utente che si è autenticato come un utente amministratore (admin) sarà in grado di:

1. Effettuare tutte le operazioni che può svolgere il conciierge
2. Inserire una nuova categoria per le recensioni/domande o disattivare una specifica categoria
3. Approvare il pagamento di un cliente relativo ad una prenotazione di una camera
4. Visualizzare i clienti attuali (compresi anche quelli non ancora presenti in struttura) e modificare i loro dati

1.2 Casi d’uso

In questa sezione verranno approfondite le funzionalità che vengono fornite a ciascun tipo d’utente.

1.2.1 Visualizzazione informazioni relative all'hotel

Essenzialmente vengono semplicemente mostrate le informazioni principali relative all'hotel quali:

1. Una breve descrizione dell'hotel e delle camere.
2. Locazione ed attività messe a disposizione (il tutto accompagnato da una serie di immagini illustrative).
3. Recensioni lasciate dai clienti.
4. Una serie di contatti.

Dalla pagina home sarà poi possibile accedere ad una pagina dedicata alle FAQ, nella quale verrà mostrato un elenco di domande che vengono effettuate frequentemente, seguite da una risposta.

1.2.2 Prenotazione di un soggiorno

Mediante un'apposita pagina dedicata è possibile prenotare un soggiorno presso l'hotel. Ciò avviene mediante l'inserimento da parte dell'utente di due date: la data di arrivo presso l'hotel e la data di ripartenza. Sulla base di queste date il sito mostrerà un elenco di camere disponibili che vengono distinte in tre categorie:

- **Camera standard singola** (prezzo pari a 30€ a notte).
- **Camera standard doppia** (prezzo pari a 60€ a notte).
- **Camera suite** (prezzo pari a 150€ a notte).

Una camera può essere messa a disposizione anche se ha già una o più prenotazioni associate ad essa (l'importante è che i periodi delle varie prenotazioni non si sovrappongano). Dopo aver selezionato una camera, se l'utente non ha già effettuato l'autenticazione, verrà richiesto di eseguire il login:

1. Se l'utente ha già effettuato la registrazione allora dovrà semplicemente inserire il proprio *username* e *password* per completare la prenotazione in tempi brevi.
2. Altrimenti, l'utente dovrà necessariamente effettuare la registrazione per completare la prenotazione.

1.2.3 Registrazione

La registrazione all'interno del sito del hotel può essere effettuata esclusivamente da un utente che non sia già registrato (e dunque che non disponga di un account). Essa può essere effettuata in due circostanze:

1. Quando un utente effettua la sua prima prenotazione di una camera. In questo caso, non avendo ancora usufruito dell'hotel dovrà necessariamente registrare un proprio account per completare la prenotazione.
2. Oppure l'utente potrà registrarsi all'interno del sito anticipatamente (e quindi con la possibilità di effettuare la prima prenotazione in un secondo momento). In questo caso l'utente potrà comunque accedere alla propria pagina personale per poter modificare i propri dati, ma non potrà ovviamente usufruire di funzionalità aggiuntive finché non prenoterà un soggiorno.

Indipendentemente dal momento in cui viene effettuata la registrazione, verranno richiesti di inserire i seguenti dati: *nome, cognome, codice fiscale, data di nascita, indirizzo, telefono* ed *email*. Una volta inseriti i dati, l'utente per poter diventare a tutti gli effetti un *cliente registrato* dovrà scegliere le proprie credenziali (*username* e *password*) con cui effettuerà il login.

1.2.4 Login

Mediante un apposita pagina l'utente è in grado di inserire le proprie credenziali (*username* e *password*) per poter effettuare l'autenticazione. Ogni tipologia di utente disporrà di un form separato per poter inserire le proprie credenziali. Ad esempio un utente amministratore per poter effettuare l'autenticazione dovrà inserire le proprie credenziali nel suo form dedicato ed inserirle in un altro form non porterà il successo del login (anche se le credenziali sono corrette).

1.2.5 Visualizzazione e modifica delle informazioni personali

Un cliente dopo che avrà effettuato l'autenticazione verrà portato in una pagina personale in cui potrà ottenere un riepilogo di tutti i dati inseriti al momento della registrazione (compresi *username* e *password*). Eventualmente, avrà la possibilità di scegliere un qualsiasi dato per poterlo modificare. Tutte queste possibilità vengono date anche ad i clienti che non hanno un

soggiorno prenotato. Inoltre, i clienti potranno visualizzare i dati relativi ad eventuali prenotazioni di soggiorni passati e presenti (senza però poterli modificare)

1.2.6 Pagamento mediante denaro e/o crediti

Un cliente avrà la possibilità di eseguire il pagamento della prenotazione di un soggiorno, di un servizio in camera o di un'attività in tre modi distinti:

- **Interamente in denaro**
- **Interamente in crediti**
- **Misto** (sia denaro che crediti)

Ovviamente, un cliente potrà pagare la prenotazione di una camera usando i crediti esclusivamente se ha già effettuato precedenti soggiorni in passato e dunque risulta essere un utente registrato da tempo (un utente che effettua la prima prenotazione non potrà pagare in crediti poiché essendosi appena registrato non ha avuto la possibilità di accumularli).

1.2.7 Servizio di ristorazione

Esso presenta una home page accessibile esclusivamente dalla pagina personale di un *cliente prenotato*. Mediante un apposito menù all'interno di questa home page, sarà possibile selezionare diverse voci per poter:

- **Visualizzare il menù del ristorante:** si verrà portati in una pagina dedicata in cui sarà possibile visualizzare il menù attualmente disponibile. Verrà mostrata dunque una lista delle portate disponibili con il loro prezzo associato.
- **Prenotare un tavolo:** si verrà portati in una pagina in cui verrà richiesto di specificare alcuni dati di prenotazione:
 1. La **data** a cui la prenotazione fa riferimento (deve essere compresa nell'intervallo temporale di soggiorno del cliente).
 2. L'**orario** a cui la prenotazione fa riferimento (deve essere compresa negli intervalli di apertura a pranzo o a cena del ristorante).
 3. La **locazione** del tavolo (che può essere *interna* o *esterna*)

Tramite un apposito pulsante sarà poi possibile confermare la propria prenotazione (nel caso in cui non vi siano tavoli disponibili che soddisfano le richieste inserite verrà mostrato un apposito messaggio d'errore).

- **Prenotare un servizio in camera:** come per la prenotazione del tavolo, verrà richiesto di inserire la **data** e l'**orario** della prenotazione. Tuttavia, verrà richiesto inoltre:
 1. Le **portate** che il cliente intende ordinare (volendo un cliente potrà ordinare anche più volte la stessa portata)
 2. Delle eventuali **richieste** riguardanti il servizio o un particolare piatto, che il cliente può specificare (ad esempio per indicare delle allergie)

Verrà poi mostrato un riepilogo delle proprie scelte ed il prezzo totale calcolato in base alle portate che sono state inserite nell'ordine. Se il cliente conferma il pagamento allora l'ordine verrà registrato.

- **Visualizzare le proprie prenotazioni:** nel caso in cui l'utente abbia già effettuato una prenotazione egli mediante questa pagina sarà in grado di ottenere un riepilogo delle proprie prenotazioni effettuate (sia del servizio al tavolo e sia del servizio in camera). Tale pagina permette anche la loro cancellazione: sarà sufficiente selezionare una prenotazione da cancellare e poi mediante un apposito pulsante la prenotazione verrà rimossa (ovviamente solo se il cliente non ha ancora usufruito della prenotazione).

1.2.8 Scrittura e commento di recensioni/commenti

Mediante un'apposita voce di un menù della pagina home sarà possibile accedere ad una pagina per poter visualizzare tutte le recensioni che sono state create da utenti clienti dell'hotel. L'utente, oltre a poter visualizzare la lista di recensioni, se ha già effettuato l'autenticazione sarà in grado di scrivere un commento ad una recensione già esistente oppure inserire una propria recensione. Ogni recensione sarà composta da:

- **Nome e cognome dell'autore**
- **Una categoria**
- **Testo della recensione.**

- **Un voto** (lasciato dall'autore; può variare da 1 a 5)
- **Giudizi lasciati da altri utenti** (la recensione mostrerà solo la somma totale)
- **Eventuali commenti di risposta** (che verranno aggiunti da altri utenti).

I commenti di risposta a loro volta saranno caratterizzati da: nome e cognome dell'utente che scrive il commento, il testo del commento, giudizi lasciati da altri utenti (analoghi a quelli delle recensioni) ed eventualmente altri commenti di risposta al commento.

Tuttavia, un cliente potrà inserire o giudicare una particolare recensione solamente se la categoria di tale recensione corrisponde ad un servizio utilizzato dal cliente. Analogamente, un cliente potrà inserire un commento o giudicare un commento già esistente solamente nell'ambito di recensioni con categorie "valide" per lo specifico cliente.

1.2.9 Attività

Dalla pagina personale di un *cliente prenotato* è possibile accedere ad una pagina dedicata alle attività che l'hotel mette a disposizione dei propri clienti. In questa pagina verrà visualizzata una lista delle attività disponibili, ciascuna avente una breve descrizione ed un'immagine illustrativa. Per ogni attività saranno inoltre visualizzati i rispettivi orari di apertura e di chiusura, un eventuale prezzo orario (alcune attività possono essere gratuite) ed un form contenente:

1. Un campo per inserire la **data** di prenotazione (deve essere compresa nell'intervallo temporale di soggiorno del cliente).
2. Due campi per poter inserire **l'orario di inizio e di fine** della prenotazione (che devono essere compresi nell'intervallo di apertura e di chiusura dell'attività che si sta prenotando).
3. Un bottone per poter confermare la propria prenotazione.

Sempre da questa pagina sarà possibile, tramite una voce di un apposito menù, visualizzare le proprie prenotazioni che eventualmente sono state già effettuate. Nel caso, l'utente potrà selezionare una prenotazione (di cui non ha ancora usufruito) e cancellarla.

1.2.10 Valutazione di recensioni e di commenti

Nella pagina dedicata alle recensioni, un cliente che ha effettuato l'autenticazione avrà la possibilità di inserire un proprio giudizio sulle recensioni e sui commenti che visualizza. In particolare, è possibile inserire due tipologie di giudizi:

- **Apprezzamento per l'utilità della recensione/commento** (voto da 1 a 5)
- **Accordo con il contenuto** (voto da 1 a 3)

Un cliente potrà inserire giudizi solo su recensioni e commenti riguardanti una categoria associata ad un servizio di cui il cliente ha usufruito.

Un cliente potrà inserire solo una volta un particolare giudizio associato ad una specifica recensione o uno specifico commento ma potrà modificare un giudizio che ha inserito precedentemente (in sostanza, non possono essere presenti due giudizi distinti dello stesso tipo, inseriti dallo stesso cliente ed associati alla stessa recensione o allo stesso commento).

1.2.11 Inserimento e risposta alle domande

Dalla propria pagina personale l'utente avrà la possibilità di accedere ad una pagina dedicata alle domande inserite dagli utenti clienti dell'hotel. Esse non sono da confondere con le FAQ:

1. Le FAQ sono accessibili direttamente dalla home page del sito e visibili a tutte le tipologie di utenti (e sono in numero ridotto)
2. Le *domande* che inseriscono i clienti dell'hotel sono visibili esclusivamente da un utente che abbia effettuato l'autenticazione (e dunque la registrazione)

Ogni domanda sarà composta da una *categoria*, un *testo* e da eventuali risposte lasciate da altri clienti dell'hotel o dal concierge. Tuttavia, un cliente dell'hotel potrà inserire delle domande o rispondere a domande già esistenti solo se la categoria della domanda fa parte di un servizio utilizzato effettivamente dal cliente.

1.2.12 Modifica del menù del ristorante

L'utente moderatore/amministratore sarà portato in una pagina in cui egli potrà scegliere la tipologia di azione che intende compiere:

- **Aggiunta:** verrà mostrato all'utente un form in cui egli dovrà inserire tutti gli attributi (*tipologia, nome e prezzo*) della portata che intende aggiungere.
- **Modifica/cancellazione:** verrà richiesto, mediante un apposito menù a tendina, di inserire la tipologia di portata (antipasti, primi piatti, ecc..) che si intende modificare/cancellare. Dopodiché, verrà mostrata la lista di portate che corrispondono alla tipologia scelta. L'utente potrà selezionare una portata e, mediante degli appositi pulsanti, decidere se intende cancellarla dal menù oppure modificarla. Nel caso in cui egli intende modificare la portata verrà data la possibilità di modificare il campo del nome e del prezzo.

1.2.13 Modifica degli orari

Verrà mostrata all'utente moderatore/amministratore una lista contenente tutte le attività organizzate dall'hotel con i loro rispettivi orari di inizio e di fine. In questa lista sarà inoltre incluso il ristorante dell'hotel con i rispettivi orari di apertura e chiusura a pranzo ed a cena. L'utente poi potrà selezionare l'attività della quale intende modificare l'orario, inserire le nuove fasce orarie e confermare la modifica mediante un apposito pulsante. Ciò vale anche per il ristorante con la differenza che l'utente potrà modificare al più **due fasce orarie** (quella del pranzo e quella della cena).

1.2.14 Modifica delle attività

All'utente moderatore/amministratore verrà data la possibilità di modificare tutte le informazioni relative ad un'attività. Verrà prima mostrata una lista contenente tutte le attività dell'hotel; dopodiché, dopo aver selezionato l'attività che si intende modificare, verrà data la possibilità di modificare il suo *nome*, la *descrizione*, *l'immagine* ed il suo *prezzo orario* (gli orari non possono essere modificati poiché la loro modifica avviene nella pagina dedicata alla modifica degli orari).

1.2.15 Visualizzazione delle prenotazioni dei clienti

Verrà mostrata al concierge (o all'admin) una pagina contenente le prenotazioni di ogni singolo cliente. Le prenotazioni sono distinte tra: prenotazioni del servizio di ristorazione e prenotazioni di attività. A prescindere dal tipo, l'utente sarà in grado di selezionare una specifica prenotazione e poi potrà scegliere se intende annullarla (e quindi cancellarla) oppure modificarla. La modifica dipende dal tipo di prenotazione che è stata selezionata:

1. Se si vuole modificare una prenotazione del servizio di ristorazione: l'admin potrà modificare il numero del tavolo (e di conseguenza anche la locazione), l'orario di prenotazione o la data di prenotazione.
2. Se si vuole modificare una prenotazione di una attività: l'admin potrà modificare l'orario di inizio o di fine della prenotazione oppure la data della prenotazione stessa.

1.2.16 Visualizzazione dei clienti attuali

Il concierge (o l'admin) verrà portato in una pagina dedicata per ottenere un *report* dei clienti che hanno prenotato un soggiorno presso l'hotel. In particolare viene visualizzata una lista che, per ogni cliente, mostra le generalità di quest'ultimo e la prenotazione che egli ha effettuato. Sono inoltre mostrati, in modo distinto da i clienti precedenti, anche i clienti che *non sono ancora presenti in struttura* (per poterli identificare la piattaforma effettua un confronto tra la data odierna e la data di inizio prenotazione). Eventualmente, l'utente potrà modificare i dati personali di uno specifico cliente.

1.2.17 Check-out di un cliente

Il check-out di un cliente serve per indicare che il soggiorno di quest'ultimo è terminato e che dunque non è più necessario tener traccia dei dati relativi al cliente. Viene richiesto all'utente amministratore di inserire il codice fiscale del cliente di cui si intende effettuare il check-out. Dopodiché, verrà visualizzata una lista contenente i dati del cliente inserito e tutte le prenotazioni effettuate da quest'ultimo. Se l'amministratore conferma di voler effettuare il check-out allora il soggiorno del cliente viene considerato *terminato* e, di conseguenza, vengono eliminati dalla piattaforma tutti i dati relativi a quest'ultimo e le prenotazioni che ha effettuato.

1.3 Strutture dati

In questa sezione verranno descritti la serie di file xml necessari per gestire i dati dell'applicazione.

1.3.1 Clienti.xml

File che servirà per memorizzare tutti i clienti dell'hotel (e quindi tutti gli utenti che hanno prenotato una camera). Per ogni utente verranno memorizzati tutti i campi richiesti al momento della prenotazione (*nome*, *cognome*, *codice fiscale*, *data di nascita* e *numero della carta di credito*) insieme alle credenziali che verranno utilizzate per l'autenticazione (*username* e *password*). Per motivi pratici, le prenotazioni di ogni cliente verranno memorizzate in altri file xml.

Clienti.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<listaClienti xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="Clienti.xsd">
  <cliente codFisc = "RSSGNN64R03E472G">
    <nome>Giovanni</nome>
    <cognome>Rossi</cognome>
    <dataDiNascita>1964-10-03</dataDiNascita>
    <numeroCarta>0000-0000-0000-0000</numeroCarta>
    <credenziali>
      <username>Camera110</username>
      <password>qwerty</password>
    </credenziali>
  </cliente>
</listaClienti>
```

Clienti.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <xsd:element name="listaClienti">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="cliente" minOccurs="0"
          maxOccurs="unbounded" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
```

```

<xsd:element name="cliente">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="nome" minOccurs="1" maxOccurs="1" />
      <xsd:element ref="cognome" minOccurs="1" maxOccurs="1" />
      <xsd:element ref="dataDiNascita" minOccurs="1" maxOccurs="1" />
      <xsd:element ref="numeroCarta" minOccurs="1" maxOccurs="1" />
      <xsd:element ref="credenziali" minOccurs="1" maxOccurs="1" />
    </xsd:sequence>
    <xsd:attribute name="codFisc" type="xsd:string use="required" />
  </xsd:complexType>
</xsd:element>

<xsd:element name="nome" type="xsd:string" />
<xsd:element name="cognome" type="xsd:string" />
<xsd:element name="dataDiNascita" type="xsd:date" />
<xsd:element name="numeroCarta">
  <xsd:simpleType>
    <xsd:restriction base="xsd:string">
      <xsd:pattern value="[0-9]{4}-[0-9]{4}-[0-9]{4}-[0-9]{4}" />
    </xsd:restriction>
  </xsd:simpleType>
</xsd:element>

<xsd:element name="credenziali">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="username" minOccurs="1" maxOccurs="1" />
      <xsd:element ref="password" minOccurs="1" maxOccurs="1" />
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

<xsd:element name="username">
  <xsd:simpleType>
    <xsd:restriction base="xsd:string">
      <xsd:pattern value="Camera[0-9]{3}" />
    </xsd:restriction>
  </xsd:simpleType>
</xsd:element>

<xsd:element name="password">

```

```
<xsd:simpleType>  
  <xsd:restriction base="xsd:string">  
    <xsd:pattern value="[a-zA-Z0-9]{6}" />  
  </xsd:restriction>  
</xsd:simpleType>  
</xsd:element>  
</xsd:schema>
```

1.3.2 Amministratori.xml

Questo file essenzialmente ha il solo scopo di conservare le credenziali degli utenti amministratori del sistema (*username* e *password*).

Amministratori.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<listaAdmin xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="Amministratori.xsd">
  <Admin>
    <username>Admin</username>
    <password>hotel123</password>
  </Admin>
</listaAdmin>
```

Amministratori.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <xsd:element name="listaAdmin">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="Admin" minOccurs="0"
          maxOccurs="unbounded" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="Admin">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="username" minOccurs="1"
          maxOccurs="1" />
        <xsd:element ref="password" minOccurs="1" maxOccurs="1"
          />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="username" type="xsd:string" />
  <xsd:element name="password" type="xsd:string" />
</xsd:schema>
```

1.3.3 Camere.xml

File che servirà per memorizzare la lista di camere che l'hotel mette a disposizione. Ogni camera sarà costituita da: un *numero* (univoco) , un campo *tipo*, un campo *prezzo* (dipendente dal tipo di camera) ed una serie di *prenotazioni* associate alla camera. Per ogni *prenotazione* viene registrato il *codice fiscale* del cliente che ha prenotato la camera, la *data di arrivo* e la *data di partenza*.

Camere.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<listaCamere xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="Camere.xsd">
  <Camera numero="C110">
    <tipo>Standard Doppia</tipo>
    <prezzo>60</prezzo>
    <listaPrenotazioni>
      <prenotazione>
        <codFisc>RSSGNN64R03E472G</codFisc>
        <dataArrivo>2022-10-28</dataArrivo>
        <dataPartenza>2022-10-31</dataPartenza>
      </prenotazione>
    </listaPrenotazioni>
  </Camera>
</listaCamere>
```

Camere.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <xsd:element name="listaCamere">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="Camera" minOccurs="1"
          maxOccurs="unbounded" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="Camera">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="tipo" minOccurs="1" maxOccurs="1" />
        <xsd:element ref="prezzo" minOccurs="1" maxOccurs="1" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
</xsd:schema>
```

```

        <xsd:element ref="listaPrenotazioni" minOccurs="1"
                    maxOccurs="1" />
    </xsd:sequence>
    <xsd:attribute name="numero" type="xsd:ID" use="required" />
</xsd:complexType>
</xsd:element>

<xsd:element name="tipo">
    <xsd:simpleType>
        <xsd:restriction base="xsd:string">
            <xsd:enumeration value="Standard Singola" />
            <xsd:enumeration value="Standard Doppia" />
            <xsd:enumeration value="Suite" />
        </xsd:restriction>
    </xsd:simpleType>
</xsd:element>

<xsd:element name="prezzo">
    <xsd:simpleType>
        <xsd:restriction base="xsd:int">
            <xsd:enumeration value="30" />
            <xsd:enumeration value="60" />
            <xsd:enumeration value="150" />
        </xsd:restriction>
    </xsd:simpleType>
</xsd:element>

<xsd:element name="listaPrenotazioni">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element ref="prenotazione" minOccurs="0"
                        maxOccurs="unbounded" />
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>

<xsd:element name="prenotazione">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element ref="codFisc" minOccurs="1" maxOccurs="1"
                        />
            <xsd:element ref="dataArrivo" minOccurs="1"
                        maxOccurs="1" />
            <xsd:element ref="dataPartenza" minOccurs="1"
                        maxOccurs="1" />
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>

```

```

<xsd:element name="codFisc" type="xsd:string" />
<xsd:element name="dataArrivo" type="xsd:date" />
<xsd:element name="dataPartenza" type="xsd:date" />
</xsd:schema>

```

1.3.4 Ristorante.xml

Questo file verrà utilizzato per conservare al suo interno gli orari di apertura e di chiusura del ristorante ed il menu. In particolare, si avrà: *ora apertura pranzo*, *ora chiusura pranzo*, *ora apertura cena*, *ora chiusura cena* ed una *lista di portate*. Ogni *portata* sarà caratterizzata da: una *tipologia* (che può essere *antipasto*, *primo piatto*, *secondo piatto* o *dolce*), una *descrizione* ed il *prezzo*.

Ristorante.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<ristoranti xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="Ristorante.xsd">
  <ristorante>
    <oraAperturaPranzo>12:00:00</oraAperturaPranzo>
    <oraChiusuraPranzo>15:00:00</oraChiusuraPranzo>
    <oraAperturaCena>19:00:00</oraAperturaCena>
    <oraChiusuraCena>23:00:00</oraChiusuraCena>
    <listaPortate>
      <portata tipologia="primo piatto">
        <descrizione>Spaghetti alla carbonara</descrizione>
        <prezzo>12</prezzo>
      </portata>
    </listaPortate>
  </ristorante>
</ristoranti>

```

Ristorante.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <xsd:element name="ristoranti">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="ristorante" minOccurs="1"
          maxOccurs="unbounded" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

```

```

<xsd:element name="ristorante">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="oraAperturaPranzo" minOccurs="1"
        maxOccurs="1" />
      <xsd:element ref="oraChiusuraPranzo" minOccurs="1"
        maxOccurs="1" />
      <xsd:element ref="oraAperturaCena" minOccurs="1"
        maxOccurs="1" />
      <xsd:element ref="oraChiusuraCena" minOccurs="1"
        maxOccurs="1" />
      <xsd:element ref="listaPortate" minOccurs="1"
        maxOccurs="1" />
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

<xsd:element name="oraAperturaPranzo" type="xsd:time" />
<xsd:element name="oraChiusuraPranzo" type="xsd:time" />
<xsd:element name="oraAperturaCena" type="xsd:time" />
<xsd:element name="oraChiusuraCena" type="xsd:time" />
<xsd:element name="listaPortate">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="portata" minOccurs="1"
        maxOccurs="unbounded" />
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

<xsd:element name="portata">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="descrizione" minOccurs="1"
        maxOccurs="1" />
      <xsd:element ref="prezzo" minOccurs="1" maxOccurs="1"
        />
    </xsd:sequence>
    <xsd:attribute name="tipologia" use="required">
      <xsd:simpleType>
        <xsd:restriction base="xsd:string">
          <xsd:enumeration value="antipasto" />
          <xsd:enumeration value="primo piatto" />
          <xsd:enumeration value="secondo piatto" />
          <xsd:enumeration value="dolce" />
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:attribute>
  </xsd:complexType>
</xsd:element>

```



```

    </xsd:complexType>
  </xsd:element>

  <xsd:element name="descrizione" type="xsd:string" />
  <xsd:element name="prezzo">
    <xsd:simpleType>
      <xsd:restriction base="xsd:int">
        <xsd:minExclusive value="0" />
      </xsd:restriction>
    </xsd:simpleType>
  </xsd:element>
</xsd:schema>

```

1.3.5 Tavoli.xml

Il file in questione memorizza tutti i tavoli che vengono messi a disposizione dal ristorante, con le relative prenotazioni associate effettuate dai clienti dell'hotel. In particolare, ogni *tavolo* avrà associato ad esso: il proprio *numero* (univoco), la *locazione* (che può essere *esterna* o *interna*), il *numero di posti* ed una *lista di prenotazioni*. Ogni *prenotazione* avrà ad esso associata: un *id univoco*, il *codice fiscale* del cliente, la *data* e l'*orario* della prenotazione.

Tavoli.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<listaTavoli xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="Tavoli.xsd">
  <tavolo numero="T2">
    <locazione>Interna</locazione>
    <listaPrenotazioni>
      <prenotazione id="PT1">
        <codFisc>RSSGNN64R03E472G</codFisc>
        <data>2022-10-30</data>
        <ora>21:30:00</ora>
      </prenotazione>
    </listaPrenotazioni>
  </tavolo>
</listaTavoli>

```

Tavoli.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <xsd:element name="listaTavoli">

```

```

    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="tavolo" minOccurs="1"
          maxOccurs="unbounded" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="tavolo">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="locazione" minOccurs="1" maxOccurs="1"
          />
        <xsd:element ref="listaPrenotazioni" minOccurs="1"
          maxOccurs="1" />
      </xsd:sequence>
      <xsd:attribute name="numero" type="xsd:ID" use="required" />
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="locazione">
    <xsd:simpleType>
      <xsd:restriction base="xsd:string">
        <xsd:enumeration value="Interna" />
        <xsd:enumeration value="Esterna" />
      </xsd:restriction>
    </xsd:simpleType>
  </xsd:element>

  <xsd:element name="listaPrenotazioni">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="prenotazione" minOccurs="0"
          maxOccurs="unbounded" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="prenotazione">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="codFisc" minOccurs="1" maxOccurs="1"
          />
        <xsd:element ref="data" minOccurs="1" maxOccurs="1" />
        <xsd:element ref="ora" minOccurs="1" maxOccurs="1" />
      </xsd:sequence>
      <xsd:attribute name="id" type="xsd:ID" use="required" />
    </xsd:complexType>
  </xsd:element>

```

```

<xsd:element name="codFisc" type="xsd:string" />
<xsd:element name="data" type="xsd:date" />
<xsd:element name="ora" type="xsd:time" />
</xsd:schema>

```

1.3.6 Attivita.xml

Questo file verrà utilizzato per memorizzare al suo interno tutte le *attività* che l'hotel mette a disposizione, con le relative prenotazioni associate. Per ogni attività si avrà: il relativo *nome* (che sarà univoco tra tutte le attività), una *descrizione*, un *link di un'immagine*, l'*ora d'apertura* e di *chiusura*, il *prezzo orario* ed una *lista di prenotazioni*. Per ogni *prenotazione* si avrà: un *id* univoco, il *codice fiscale* del cliente, la *data*, l'*ora d'inizio* e di *fine* prenotazione e il *prezzo totale* (dipendente dal prezzo orario dell'attività e dal numero di ore della prenotazione).

Attivita.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<listaAttivita xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="Attivita.xsd">
  <attivita>
    <nome>Palestra</nome>
    <descrizione>
      Ubicata all'interno dell'hotel ed equipaggiata con le ultime
      macchine di Technogym, una palestra interna rappresenta la
      soluzione perfetta per mantenersi in forma anche se lontani
      da casa. Hotel Sapienza vi propone una palestra dotata di:
      <![CDATA[<br />]]>Tapis roulant e cyclette<![CDATA[
      <br />]]>Materassini e ball per aerobica e pilates
      <![CDATA[<br />]]>Pesi e bilancieri di varie dimensioni
      <![CDATA[<br />]]>. Vengono inoltre messi a disposizione
      gli spogliatoi dotati di armadietti e docce
    </descrizione>
    <linkImmagine>https://www.destinazioneavventura.it/wp-content/
      uploads/2019/01/Excelsior-Sport-Hotel-Gala-Milano-Vacanza
      -Sportive.jpg</linkImmagine>
    <oraApertura>06:00:00</oraApertura>
    <oraChiusura>22:00:00</oraChiusura>
    <prezzoOrario>10</prezzoOrario>
    <listaPrenotazioni>
      <prenotazione id="PA1">
        <codFisc>RSSGNN64R03E472G</codFisc>
        <data>2022-10-29</data>
      </prenotazione>
    </listaPrenotazioni>
  </attivita>
</listaAttivita>

```

```

        <oraInizio>17:00:00</oraInizio>
        <oraFine>19:00:00</oraFine>
        <prezzoTotale>20</prezzoTotale>
    </prenotazione>
</listaPrenotazioni>
</attivit a>
</listaAttivit a>

```

Attivita.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema" >
    <xsd:element name="listaAttivit a" >
        <xsd:complexType>
            <xsd:sequence>
                <xsd:element ref="attivit a" minOccurs="1"
                    maxOccurs="unbounded" />
            </xsd:sequence>
        </xsd:complexType>
    </xsd:element>

    <xsd:element name="attivit a" >
        <xsd:complexType>
            <xsd:sequence>
                <xsd:element ref="nome" minOccurs="1" maxOccurs="1" />
                <xsd:element ref="descrizione" minOccurs="1"
                    maxOccurs="1" />
                <xsd:element ref="linkImmagine" minOccurs="1"
                    maxOccurs="1" />
                <xsd:element ref="oraApertura" minOccurs="1"
                    maxOccurs="1" />
                <xsd:element ref="oraChiusura" minOccurs="1"
                    maxOccurs="1" />
                <xsd:element ref="prezzoOrario" minOccurs="1"
                    maxOccurs="1" />
                <xsd:element ref="listaPrenotazioni" minOccurs="1"
                    maxOccurs="1" />
            </xsd:sequence>
        </xsd:complexType>
    </xsd:element>

    <xsd:element name="nome" type="xsd:ID" />
    <xsd:element name="descrizione" type="xsd:string" />
    <xsd:element name="linkImmagine" type="xsd:string" />
    <xsd:element name="oraApertura" type="xsd:time" />
    <xsd:element name="oraChiusura" type="xsd:time" />
    <xsd:element name="prezzoOrario" >
        <xsd:simpleType>

```

```

        <xsd:restriction base="xsd:double">
            <xsd:minExclusive value="0" />
        </xsd:restriction>
    </xsd:simpleType>
</xsd:element>

<xsd:element name="listaPrenotazioni">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element ref="prenotazione" minOccurs="0"
                maxOccurs="unbounded" />
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>

<xsd:element name="prenotazione">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element ref="codFisc" minOccurs="1" maxOccurs="1"
                />
            <xsd:element ref="data" minOccurs="1" maxOccurs="1" />
            <xsd:element ref="oraInizio" minOccurs="1" maxOccurs="1"
                />
            <xsd:element ref="oraFine" minOccurs="1" maxOccurs="1"
                />
            <xsd:element ref="prezzoTotale" minOccurs="1"
                maxOccurs="1" />
        </xsd:sequence>
        <xsd:attribute name="id" type="xsd:ID" use="required" />
    </xsd:complexType>
</xsd:element>

<xsd:element name="codFisc" type="xsd:string" />
<xsd:element name="data" type="xsd:date" />
<xsd:element name="oraInizio" type="xsd:time" />
<xsd:element name="oraFine" type="xsd:time" />
<xsd:element name="prezzoTotale">
    <xsd:simpleType>
        <xsd:restriction base="xsd:double">
            <xsd:minExclusive value="0" />
        </xsd:restriction>
    </xsd:simpleType>
</xsd:element>
</xsd:schema>

```

1.3.7 Recensioni.xml

Poiché l'hotel mette a disposizione la possibilità di visualizzare ed eventualmente scrivere delle recensioni, sarà necessario memorizzare tutte le recensioni che vengono scritte dai clienti dell'hotel. Per ogni *recensione* si avrà: un *id* univoco, il *nome* e *cognome* del cliente che ha scritto la recensione, un *testo*, un *voto* (compreso tra 1 e 5) ed una *lista di commenti di risposta*. Per ogni *commento* si avrà: il *nome* e *cognome* del cliente che ha scritto il commento ed il *testo* del commento.

Recensioni.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<listaRecensioni xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="Recensioni.xsd">
  <recensione id="R1">
    <nomeAutore>Giovanni</nomeAutore>
    <cognomeAutore>Rossi</cognomeAutore>
    <testoRecensione>
      Soggiorno fantastico! Il ristorante offre sempre piatti di
      immensa qualita' ed il personale e' estremamente professionale
      e sempre disponibile.
    </testoRecensione>
    <voto>5</voto>
    <listaCommenti>
      <commento>
        <nomeAutoreCommento>Luigi</nomeAutoreCommento>
        <cognomeAutoreCommento>Verdi</cognomeAutore
        Commento>
        <testoCommento>Concordo pienamente!</testoCommento>
      </commento>
    </listaCommenti>
  </recensione>
</listaRecensioni>
```

Recensioni.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <xsd:element name="listaRecensioni">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="recensione" minOccurs="0"
          maxOccurs="unbounded" />
      </xsd:sequence>
    </xsd:complexType>
```

```

</xsd:element>

<xsd:element name="recensione">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="nomeAutore" minOccurs="1"
        maxOccurs="1" />
      <xsd:element ref="cognomeAutore" minOccurs="1"
        maxOccurs="1" />
      <xsd:element ref="testoRecensione" minOccurs="1"
        maxOccurs="1" />
      <xsd:element ref="voto" minOccurs="1" maxOccurs="1" />
      <xsd:element ref="listaCommenti" minOccurs="1"
        maxOccurs="1" />
    </xsd:sequence>
    <xsd:attribute name="id" type="xsd:ID" use="required" />
  </xsd:complexType>
</xsd:element>

<xsd:element name="nomeAutore" type="xsd:string" />
<xsd:element name="cognomeAutore" type="xsd:string" />
<xsd:element name="testoRecensione" type="xsd:string" />
<xsd:element name="voto">
  <xsd:simpleType>
    <xsd:restriction base="xsd:int">
      <xsd:minInclusive value="1" />
      <xsd:maxInclusive value="5" />
    </xsd:restriction>
  </xsd:simpleType>
</xsd:element>

<xsd:element name="listaCommenti">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="commento" minOccurs="0"
        maxOccurs="unbounded" />
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

<xsd:element name="commento">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="nomeAutoreCommento" minOccurs="1"
        maxOccurs="1" />
      <xsd:element ref="cognomeAutoreCommento" minOccurs="1"
        maxOccurs="1" />
      <xsd:element ref="testoCommento" minOccurs="1"
        maxOccurs="1" />
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

```

```
</xsd:sequence>
</xsd:complexType>
</xsd:element>

<xsd:element name="nomeAutoreCommento" type="xsd:string" />
<xsd:element name="cognomeAutoreCommento" type="xsd:string" />
<xsd:element name="testoCommento" type="xsd:string" />
</xsd:schema>
```

1.4 Elementi XML

In questa sezione verranno analizzati dieci elementi XML scelti a campione, per poter giustificare la loro presenza nei vari file XML appena presentati.

Clienti.xml: credenziali

Come detto più volte nel corso di questo capitolo e descritto nella sezione **1.2.3**, un utente che prenota una camera avrà bisogno di credenziali personali per poter effettuare il login nella piattaforma. Lo scopo di questo elemento è dunque quello di incapsulare al suo interno, mediante l'utilizzo di altri due sottoelementi *username* e *password*, le credenziali che vengono assegnate ad un utente al completamento della prenotazione.

Camere.xml: tipo

Nella sezione **1.2.2** nel descrivere la funzionalità di prenotazione di un soggiorno, è stato messo in evidenza che le camere vengono distinte tra loro in base alla loro *tipologia* e, in base a quest'ultima, viene assegnato loro un prezzo fisso. L'elemento **tipo** viene dunque inserito per poter identificare le tipologie di camere ed assegnare correttamente il prezzo corrispondente.

Camere.xml: listaPrenotazioni

Poiché nella sezione **1.2.2** viene specificato che una camera può avere anche più di una prenotazione associata ad essa, questo elemento avrà il compito di incapsulare al suo interno tutte le prenotazioni associate ad una specifica camera.

Camere.xml: codFisc

Nell'introduzione del capitolo viene detto che un cliente dell'hotel, dopo che ha effettuato il login, verrà portato ad una pagina personale in cui potrà vi-

sualizzare un riepilogo del proprio soggiorno. La presenza di questo elemento è dunque giustificata dalla necessità di dover associare correttamente la prenotazione di una camera al cliente che l'ha effettuata (per poter poi reperire i dati corretti). In particolare, tutti gli altri elementi omonimi presenti nelle prenotazioni che si trovano in altri file XML, avranno lo stesso scopo (cioè quello di associare correttamente la prenotazione).

Ristorante.xml: listaPortate

Nella sezione **1.2.4** viene detto che dalla home page del servizio di ristorazione sarà possibile visualizzare il menù del ristorante. Questo elemento avrà dunque lo scopo di contenere al suo interno tutto il menù con le relative portate che vengono messe a disposizione.

Ristorante.xml: tipologia

La presenza di questo attributo è giustificata da ciò che viene detto nella sezione **1.2.9**. Infatti, l'admin dovrà prima di tutto inserire la tipologia della quale fa parte la portata che intende modificare/cancellare. Inoltre, per poter inserire una nuova portata, l'admin dovrà specificare la tipologia di quest'ultima. La presenza di questo elemento inoltre, permette una corretta suddivisione delle portate durante l'elencazione del menù.

Tavoli.xml: id

Come detto nelle sezioni **1.2.4** e **1.2.11**, sia il cliente che l'admin avranno la possibilità di visualizzare tutte le prenotazioni che sono state effettuate per ogni tavolo: il cliente può annullarle, l'admin potrà anche modificarle. A prescindere dell'azione che si intende compiere, in entrambi i casi quando un utente selezionerà una prenotazione sarà necessario identificarla univocamente per poterla ritrovare correttamente all'interno del file XML e compiere l'azione richiesta. Poiché un tavolo può avere più prenotazioni associate ad esso, l'attributo **id** avrà dunque lo scopo di garantire tale univocità.

Tavoli.xml: ora

Come detto nella sezione **1.2.4**, il cliente dovrà inserire l'orario nel quale intende prenotare il tavolo. Lo scopo principale di questo elemento è dunque quello di conservare tale informazione. Tuttavia, non è l'unico: infatti, nel momento in cui si andrà a verificare se un tavolo può essere assegnato o meno ad una prenotazione, il controllo del *range* dell'orario diventa fondamentale.

Per esempio, se un cliente intende prenotare un tavolo ad un orario appartenente al range orario del pranzo, un tavolo che è stato prenotato per l'ora di cena dello stesso giorno deve essere considerato disponibile per soddisfare la prenotazione del pranzo (ovviamente deve valere anche il viceversa).

Attivita.xml: prezzoTotale

Nella sezione **1.2.6** viene detto che un cliente, per poter effettuare la prenotazione di un attività, dovrà necessariamente inserire l'ora di inizio e di fine prenotazione. Però, poiché ciascuna attività è caratterizzata da un prezzo orario, il prezzo di ogni prenotazione è *variabile*. Questo elemento avrà dunque lo scopo di registrare al suo interno il prezzo corretto di ogni prenotazione effettuata per ogni attività. In particolare, tale prezzo dipenderà dall'ora di inizio e di fine prenotazione e dal prezzo orario della specifica attività (il calcolo verrà effettuato a livello applicativo nella pagina di prenotazione).

Recensioni.xml: id

Lo scopo di questo attributo è quello di identificare univocamente la singola recensione. Infatti, nella sezione **1.2.5** viene detto che un cliente avrà la possibilità di inserire un commento in una recensione già esistente. Tuttavia, poiché possono essere presenti più recensioni, diventa necessario identificare univocamente la singola recensione che il cliente intende commentare per poter associare correttamente il commento ad essa.