**Progetto “PollWeb” – Sviluppo Web Avanzato**

Introduzione

Il seguente progetto estende la traccia del progetto PollWeb di Ingegneria del Web di cui è stata utilizzata la sua base di dati su cui viene affiancato un servizio web RESTful che esporrà questi dai e alcune delle relative funzionalità di accesso e modifica. Non disponendo di un client è stata utilizzata l’applicazione Postman per testarne la correttezza.

Il servizio RESTful è stato sviluppato utilizzando il micro-framework lumen per PHP. L’autenticazione degli utenti non essendo persistente viene passata nell’header tramite un token presente nella tabella dell’utente che vuole effettuare una data operazione e viene gestita da un middleware di lumen. La password inserita in fase di registrazione viene criptata con una funzione hash e decriptata quando si cerca di accedere. Il Token viene creato insieme all’utente ed è generato come una stringa random di 60 caratteri e criptata in base64.

Ogni URL è preceduto dal dominio (nei test utilizzo localhost:8080) e da api/v1 che precede tutte le chiamate REST ad eccezione del login, signin e logout.

Specifiche

**URL: login**

*Verbo: POST*

Body:

{

"email": "giogio@gmail.com",

"password": "password123"

}

Output:

{

"token": "UzBNaFo0Zk1YMFRSM0FBRnVFbFI2MlFPVGg1TVVtME4xbWlDOFRzUmhENW5JVXFzZEd2M2dKZkVrcWtY"

}

Descrizione:

*Prende in input un oggetto contenente email e password di un utente e restituisce un oggetto contenente il relativo token di sessione/autenticazione.*

**URL: signin**

*Verbo: POST*

Body:

{

"name": "Johnatan",

"surname": "Joestar",

"email": "JoJo@joestar.com",

"password": "password123"

}

Output:

{

"name": "Johnatan",

"surname": "Joestar",

"email": "JoJo@joestar.com",

"role": "USER",

"id": 7

}

Descrizione:

*Prende in input un oggetto contenente nome, cognome, email e password e crea un utente e restituisce un oggetto contenente i dati forniti in input eccetto la password e il ruolo impostato di default a ‘USER’.*

**URL: logout**

Verbo: POST

Header:

api\_token: N3hGOVhHU083Q2ZpWVF1S2lyS0J4TzhLejFhS09RZDlseTFJeXdoTGdHNGlDRXFHTTFHMHJCUkFwU1F1

Output:

{

"message": "Logout eseguito con successo!"

}

Descrizione:

*Se l’utente è autenticato e dispone quindi di un api\_token, viene effettuato il logout e aggiornato il capo nel database che contiene il token con uno nuovo.*

**URL: updateRole**

Verbo: PUT

Header:

api\_token: UzBNaFo0Zk1YMFRSM0FBRnVFbFI2MlFPVGg1TVVtME4xbWlDOFRzUmhENW5JVXFzZEd2M2dKZkVrcWtY

Body:

{

"email": "JoJo@joestar.com",

"role": "RES"

}

Output:

{

"message":"Il ruolo è stato aggiornato con successo in: RES"

}

Descrizione:

Se si è autenticati come ‘ADMIN’ si può modificare il ruolo di un utente che viene selezionato tramite il campo ‘email’ presente nel body.

**URL: poll**

Verbo: GET

Output:

[

{

"title": "Sondaggio #1",

"URLPoll": "http://localhost:8000/api/v1/poll/"

},

{

"title": "Sondaggio #2",

"URLPoll": "http://localhost:8000/api/v1/poll/2"

},

…

…

]

Descrizione:

*Restituisce un oggetto contenente i titoli di tutti i sondaggi nel sistema e la URI per visualizzarne i dettagli.*

**URL: poll/{id}**

Verbo: GET

Body:

Output:

{

"id": 5,

"idNum": "OlvUDrksTFQE2lX3",

"title": "Valutazione esperienza Google",

"openText": "Testo di apertura",

"closeText": "Testo di chiusura",

"openPoll": 0,

"statePoll": 0,

"URLPoll": "http://localhost:8000/api/v1/poll/5",

"user": {

"id": 4,

"name": "Giorno",

"surname": "Giovanna",

"email": "giogio@gmail.com",

"role": "ADMIN"

}

}

Descrizione:

*Restituisce un oggetto contenente tutti i dettagli relativi al sondaggio identificato da {id}.*

**URL: poll**

Verbo: POST

Header:

api\_token: UzBNaFo0Zk1YMFRSM0FBRnVFbFI2MlFPVGg1TVVtME4xbWlDOFRzUmhENW5JVXFzZEd2M2dKZkVrcWtY

Body:

{

"title": "Scelta template",

"openText": "compila il sondaggio",

"closeText": "grazie per la partecipazione",

"openPoll": "1",

"statePoll": "0",

}

Output:

{

"title": "Scelta template",

"openText": "compila il sondaggio",

"closeText": "grazie per la partecipazione",

"openPoll": "1",

"statePoll": "0",

"idNum": "5bOLSN7i7gkkCpd8",

"URLPoll": "http://localhost:8000/api/v1/poll/11",

"id": 11

}

Descrizione:

*Prende in input i dati che compongono il sondaggio (titolo, testo di apertura e di chiusura, pubblico/privato, compilabile/chiuso) gli altri dati vengono aggiunti in automatico dal controller tra cui anche l’URI che viene utilizzato per visualizzare i dettagli del sondaggio. Se l’utente che inserisce il sondaggio ha ruolo ‘USER’ viene restituito un errore e 401 Unauthorized (viene considerato come non autenticato anziché restituire 403 Forbidden). Tramite la Validation di lumen, vengono controllati tutti i dati in input della richiesta e si controlla che siano presenti quelli richiesti, altrimenti viene restituito un errore e 400 Bad Request.*

**URL: poll/{id}**

Verbo: PUT

Header:

api\_token: UzBNaFo0Zk1YMFRSM0FBRnVFbFI2MlFPVGg1TVVtME4xbWlDOFRzUmhENW5JVXFzZEd2M2dKZkVrcWtY

Body:

{

"title": “nuovo titolo”

}

Output:

{

"updated": true

}

Descrizione:

*Modifica le informazioni relative ai dati del sondaggio identificato da {id}. Nell’header è presente l’API token dell’utente autenticato, permette alla richiesta di identificarlo e valutare se è possibile eseguirla. Se l’utente è ‘ADMIN’ può modificare qualunque sondaggio, se l’utente è lo stesso che ha inserito il sondaggio può modificarlo; altrimenti viene restituito un errore e 403 Forbidden. Se tra i sondaggi non è presente quello identificato da {id} viene restituito un errore e 404 Not Found.*

**URL: poll/{id}**

Verbo: DELETE

Header:

api\_token: UzBNaFo0Zk1YMFRSM0FBRnVFbFI2MlFPVGg1TVVtME4xbWlDOFRzUmhENW5JVXFzZEd2M2dKZkVrcWtY

Output:

{

"deleted": true

}

Descrizione:

*Elimina i dati del sondaggio identificato da {id}. I controlli vengono effettuati nello stesso modo della put.*

**URL: poll/{idpoll}/question**

Verbo: GET

Output:

{

"1": "e'questo il sondaggio 7?",

"2": "si può cambiare la posizione?",

"3": "valuta questa funzione",

"4": "domanda modificata",

"5": "queste domande servono per fare numero?"

}

Descrizione:

*Restituisce oggetto contenente i dati importanti delle domande (titolo e posizione) del sondaggio identificato da {idpoll}. Se non sono presenti domande viene restituito un messaggio di errore e il codice di stato 404 Not Found.*

**URL: poll/{idpoll}/ question/{id}**

Verbo: GET

Output:

{

"positionNumber": 1,

"uniqueCode": "ggqUaICjpEutlBDG",

"questionText": "e'questo il sondaggio 7?",

"note": "NONE",

"mandatory": 1,

"questionType": "SINGLECHOISE",

"minimum": "NONE",

"maximum": "NONE",

"questionOption": "si $!br# no"

}

Descrizione:

*Restituisce un oggetto che contiene i dati dettagliati della domanda identificata dal sondaggio {idpoll} e dal suo identificativo {id}. Se non viene trovata la domanda viene restituito un messaggio di errore e 404 Not Found.*

**URL: poll/{idpoll}/question**

Verbo: POST

Header:

api\_token: UzBNaFo0Zk1YMFRSM0FBRnVFbFI2MlFPVGg1TVVtME4xbWlDOFRzUmhENW5JVXFzZEd2M2dKZkVrcWtY

Body:

{

"questionText": "quante sono le domande?",

"questionType": "SINGLECHOISE",

"questionOption": "tante $!br# tante come seconda scelta"

}

Output:

{

"questionText": "quante sono le domande?",

"questionType": "SINGLECHOISE",

"questionOption": "tante $!br# tante come seconda scelta",

"uniqueCode": "iirgzXEzH58YODsd",

"positionNumber": 6

}

Descrizione:

*Prende in input i dati che compongono la domanda (testo, tipo e opzioni domanda se sono presenti), gli altri dati vengono aggiunti in automatico dal controller. L’utente è autenticato nell’header e come avviene anche per la POST dei sondaggi; se l’utente che inserisce il sondaggio ha ruolo ‘USER’ viene restituito un errore e 401 Unauthorized (viene considerato come non autenticato anziché restituire 403 Forbidden). Tramite la Validation di lumen, vengono controllati tutti i dati in input della richiesta e si controlla che siano presenti quelli richiesti, altrimenti viene restituito un errore e 400 Bad Request.*

**URL: poll/{idpoll}/question/{id}**

Verbo: PUT

Header:

api\_token: UzBNaFo0Zk1YMFRSM0FBRnVFbFI2MlFPVGg1TVVtME4xbWlDOFRzUmhENW5JVXFzZEd2M2dKZkVrcWtY

Output:

{

"updated": true

}

Descrizione:

*Similmente a quanto visto prima, modifica le informazioni relative ai dati della domanda identificata da {id} appartenente al sondaggio identificato da {idpoll}. Nell’header è presente l’API token dell’utente autenticato, permette alla richiesta di identificarlo e valutare se è possibile eseguirla. Se l’utente è ‘ADMIN’ può modificare qualunque sondaggio, se l’utente è lo stesso che ha inserito il sondaggio può modificarlo; altrimenti viene restituito un errore e 403 Forbidden. Se tra i sondaggi non è presente quello identificato da {id} viene restituito un errore e 404 Not Found.*

**URL: poll/{idpoll}/question/{id}**

Verbo: DELETE

Header:

api\_token: UzBNaFo0Zk1YMFRSM0FBRnVFbFI2MlFPVGg1TVVtME4xbWlDOFRzUmhENW5JVXFzZEd2M2dKZkVrcWtY

Output:

{

"delated": true

}

Descrizione:

*Elimina i dati del sondaggio identificato da {id} appartenente al sondaggio {idpoll}. I controlli vengono effettuati nello stesso modo della put.*

**URL: poll/{idpoll}/question/{idquestion}/answer**

Verbo: GET

Header:

api\_token: UzBNaFo0Zk1YMFRSM0FBRnVFbFI2MlFPVGg1TVVtME4xbWlDOFRzUmhENW5JVXFzZEd2M2dKZkVrcWtY

Output:

[

"8: Holy grail?",

[

{"answer":"1"},

{"answer":"1"},

{"answer":"2"},

{"answer":"1"}

]

]

Descrizione:

*Se l’utente è autenticato viene restituito il nome della domanda e le risposte(anonime), le risposte sono delle stringhe e se sono numeri solitamente vengono gestiti dal client a seconda del tipo di domanda (risposta singola, multipla etc.).*

**URL: poll/{idpoll}/instance**

Verbo: GET

Header:

api\_token: UzBNaFo0Zk1YMFRSM0FBRnVFbFI2MlFPVGg1TVVtME4xbWlDOFRzUmhENW5JVXFzZEd2M2dKZkVrcWtY

Output:

[

"sondaggio modificato",

[

{

"name": "Giorno",

"surname": "Giovanna",

"email": "giogio@gmail.com",

"userStatus": 0,

"submission": "2019-08-13 20:38:50"

},

...

...

{

"name": "Jotaro",

"surname": "Kujo",

"email": "JoJo@bjoestar.com",

"userStatus": 0,

"submission": "2019-08-15 18:40:02"

}

]

]

Descrizione:

*Se l’utente è autenticato e dispone dei permessi necessari vengono restituite le istanze del sondaggio identificato da {idpoll} in cui sono presenti gli utenti che sono iscritti ad un sondaggio con i propri dati, se è possibile per loro partecipare (a seconda che il sondaggio sia stato chiuso o se lo hanno già compilato) e l’ultima azione effettuata).*

**URL: poll/{idpoll}/instance**

Verbo: POST

Header:

api\_token: UzBNaFo0Zk1YMFRSM0FBRnVFbFI2MlFPVGg1TVVtME4xbWlDOFRzUmhENW5JVXFzZEd2M2dKZkVrcWtY

Body:

{

"IDuser":"5"

}

Output:

{"message":"Instanza creata con successo"}

Descrizione:

*Se l’utente è autenticato e dispone dei permessi necessari per effettuare modifiche aggiunge una nuova istanza a un sondaggio associandogli un utente.*