Chiavari FrontOffice API

Documentazione tecnica delle API per il FrontOffice

Fornitore	*Moveax S.r.l.*
Cliente	*Comune di Chiavari*

Revisioni

Versione	Descrizione	Autore	Data
0.1-DRAFT	Prima Stesura	Gianluca De Cicco	12/02/2020
0.2	Definizione politiche di Login	Valerio Cervo	30/03/2020

Indice

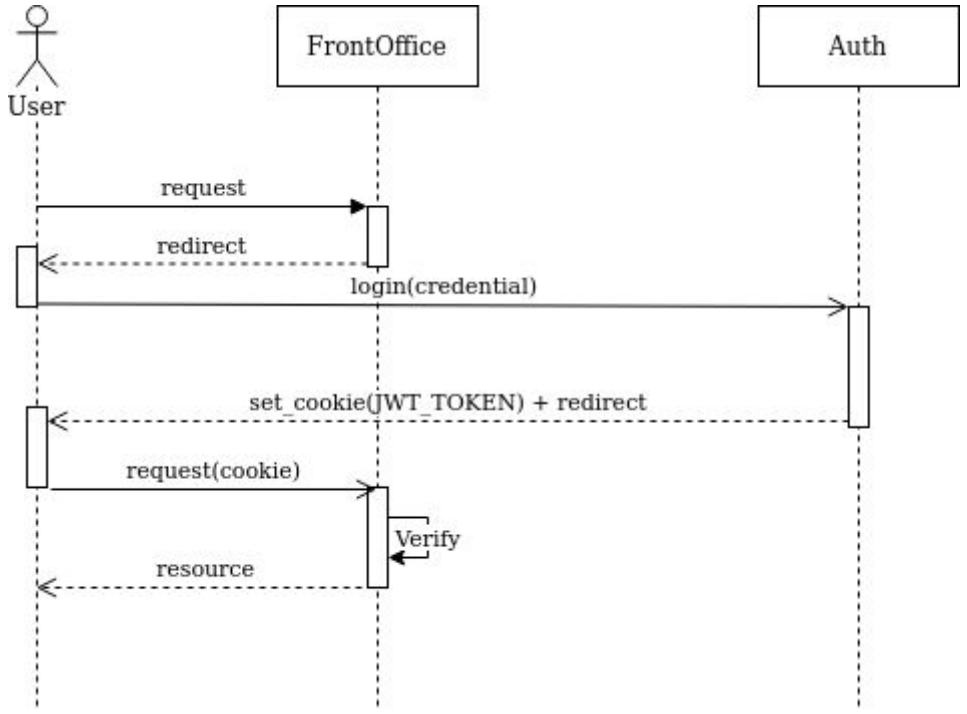
Overview

E’ qui documentato il flusso di autenticazione utente per il Frontoffice e le API esposte per la gestione pratiche.
Il flusso di integrazione prevede che un utente si autentichi tramite il portale di autenticazione, accede al Frontoffice ed ha qui la possibilità, da una parte di sottomettere una nuova pratica, con i relativi documenti; dall’altra rivedere ed eventualmente aggiornare, le pratiche da lui sottomesse.

User Authentication

Il front-office è protetto da autenticazione tramite **JWT Token** che viene rilasciato dal servizio di autenticazione attraverso il seguente User Flow e salvato come *cookie*. Il token rilasciato deve essere verificarlo rispetto alla chiave pubblica del servizio di autenticazione.

User Flow



User Login

Per verificare che l'utente sia registrato e loggato nel sistema è necessario, al primo accesso al frontoffice, effettuare un redirect al seguente URL:

```
GET
https://login.digitale.comune.chiavari.ge.it?redirect_url={redirectUrl}
```

Dove *redirectUrl* è l'URL al quale redirezionare l'utente una volta effettuato il login, ovvero alla pagina inizialmente richiesta dall'utente al primo accesso.

L'utente verrà quindi redirezionato alla pagina di login dove potrà effettuare il login o registrarsi al sistema. Al termine della procedura di login verrà rilasciato un cookie **comune_chiavari_ge_it_idtoken** contenente il token JWT che identifica l'utente attraverso le piattaforme.

Formato del token

Il token contiene informazioni relative all'utente e può essere utilizzato per recuperare alcuni dati:

```
{
  "iss": "comune.chiavari.ge.it",
  "nbf": 1586511058,
  "iat": 1586511058,
  "exp": 1586514658,
  "sub": "d1b0e3d7-98dc-404f-b3e8-15a0709f0f70",
  "user": {
    "id": "d1b0e3d7-98dc-404f-b3e8-15a0709f0f70",
    "first_name": "Mario",
    "last_name": "Rossi",
    "email": "mario.rossi@example.com",
    "profile": {
      "first_name": "Mario",
      "last_name": "Rossi",
      "full_name": "Mario Rossi",
      "professional_title": {
        "long": "Architetto",
        "short": "Arch."
      },
      "address": {
        "street_name": "Via del corso 113",
        "postcode": "00100",
        "city": "Roma",
```

```
        "province": "Roma",
        "country": "IT"
      },
      "gender": "M",
      "fiscal_code": "DNCCCLSSLSKKA",
      "birth_date": "23/05/2000",
      "birthplace": {
        "city": "H501",
        "province": "RM",
      },
      "pec": null
    }
  },
  "type": "user"
}
```

Verifica del token

Per assicurarsi che il token non sia contraffatto è necessario verificare che sia stato firmato dal server utilizzando la chiave pubblica fornita. A questo link è possibile trovare tools per la verifica e la decodifica del token in diversi linguaggi e per diversi frameworks: https://jwt.io/

User Logout

Per effettuare il logout di un utente è sufficiente redirezionare l'utente tramite una richiesta POST al seguente url:

```
POST
https://login.digitale.chiavari.ge.it/logout?redirect_url={redirectUrl}
```

Dove *redirectUrl* L'URL al quale redirezionare l'utente una volta effettuato il logout.

Frontoffice API

Le api sono raggiungibili ai seguenti URL in base all'ambiente scelto:

Ambiente	URL
Staging	-
Produzione	https://api.digitale.comune.chiavari.ge.it/

Service Authentication

Gli endpoint esposti per il frontoffice, per la sottomissione di pratiche sono protette da autenticazione con **Bearer Token**. Ad ogni richiesta è necessario specificare il token impostando un header HTTP come segue:

```
Authorization: Bearer <token>
```

Nel caso il token risulti non valido il server risponderà con un codice *HTTP* **401 Unauthorized**

Login

L'ottenimento di un token avviene tramite autorizzazione **Basic,** occorre quindi impostare l'header HTTP *Authorization* con seguente stringa:

```
Authorization: Basic <base64(client_id:secret)>
```

Dove *client_id* e *secret* sono le credenziali fornite per eseguire l'accesso. Tale richiesta deve essere inviata al seguente endpoint:

```
POST
/auth/token
```

Nel caso le credenziali non risultino valide o mal formattate il server risponderà con un codice HTTP **401 Unauthorized**

Response

```
{
    "token": {
        "value": "eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJp...",
        "expires_in": 3600
    },
    "refresh_token": {
        "value": "o2kw6CtDzQ7Y...",
        "expires_in": 2592000
    }
}
```

In risposta verranno ritornati rispettivamente il token di autenticazione e il *refresh token* il quale può essere utilizzato per ottenere un nuovo token senza dover ripetere la procedura di login. I valori nei campi *expires_in* esprimono rispettivamente la validità in secondi del token, ad esempio il token ha una validità di 60 minuti.

Refresh di un token

Il seguente endpoint consente il rilascio di un token utilizzando il *Refresh Token* ottenuto durante la fase di login:

```
POST
/auth/refresh
```

Request

```
{
      "refresh_token": "o2kw6CtDzQ7Y..."
}
```

Response

```
{
    "token": {
        "value": "eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJp...",
        "expires_in": 3600
    },
    "refresh_token": {
        "value": "Y2DB9mYm...",
        "expires_in": 2592000
    }
}
```

Creazione Pratica

Il seguente endpoint consente la sottomissione di una nuova pratica

```
POST
/frontoffice/pratiche
```

Request

```
{
  ‘category’: <string>, # categoria della pratica (eg. dehors)
  ‘referente’:{
  },
  ‘tecnico’: {
  },
  …
}
```

Response

```
{
  ‘id’: <UUID>
}
```

Dettaglio pratica

Il seguente endpoint consente di ricevere il dettaglio di una pratica sottomessa

```
GET
/frontoffice/pratiche/<id>
```

Response

```
{
  ‘category’: <string>, # categoria della pratica (eg. dehors)
  ‘referente’:{
  },
  ‘tecnico’: {
  },
  ‘state’: <Enum>,

   …
```

```
  ‘documents’: [
    {
      ‘fileName’: <string>,
      ‘id’: <UUID>
    },

    …
  ]
}
```

Modifica Pratica

Il seguente endpoint consente di modificare una nuova pratica se in un ostato consono

```
PATCH
/frontoffice/pratiche/<id>
```

Request

```
{
 …
}
```

Response

```
{
  ‘id’: <UUID>,
}
```

Una pratica può essere modificata solo se in determinati stati.
Se si prova a modificare una pratica non modificabile, viene restituito un errore

Upload Documenti Pratica

Il seguente endpoint consente il caricamento di un nuovo documento per una pratica.
Un documento può essere caricato solo se la pratica è modificabile.

```
POST
/frontoffice/pratiche/<id>/documents
```

Request

```
json | form-data
```

Response

```
{
  ‘id’: <UUID>
}
```

Se la pratica non è modificabile, viene restituito un errore

Dettaglio Documento Pratica

Il seguente endpoint consente il ricevere il documento di pratica in *base64*.

```
GET
/frontoffice/pratiche/<id_pratica>/documents/<id>
```

Response

```
{
  ‘id’: <UUID>,
  ‘fileName’: <string>,
  ‘data’: <base64 encoded file>
}
```

Rimozione Documento Pratica

Il seguente endpoint consente di eliminare un documento associato ad una pratica.
Un documento può essere eliminato solo se la pratica è modificabile.

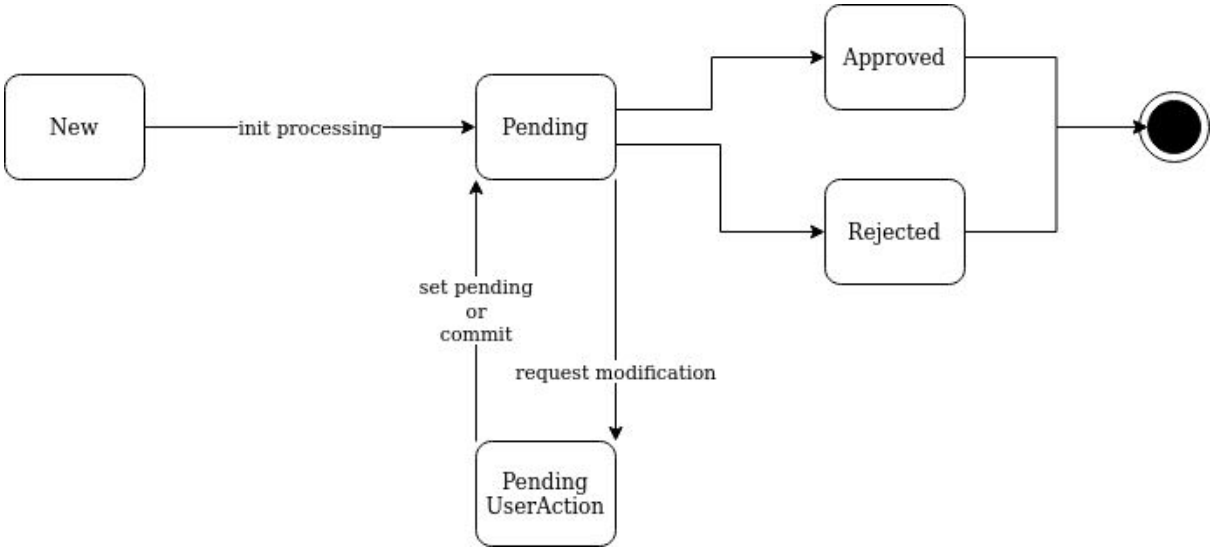
```
DELETE
/frontoffice/pratiche/<id_pratica>/documents/<id>
```

Response

```
{
  ‘id’: <UUID>,
  ‘fileName’: <string>,
  ‘data’: <base64 encoded file>
}
```

Stati di una pratica

Di seguito è riportata una bozza preliminare dei possibili stati di una pratica:



Una pratica è modificabile solo negli stati *New* e *PendinfUserAction*.