



Sommario

Manuale Sistema Gestione e verifica Pass	2
Storia del documento	2
Acronimi	2
Scopo del documento	2
Contesto	2
Architettura	2
Installazione	2
Installazione in ambiente Linux con nginx	4
Manuale funzionale	5
Creazione del modello	6
Creazione della serie e sottoserie	7
Verifica del pass	9
Ruoli utente	10

Manuale Sistema Gestione e verifica Pass

Storia del documento

DATA	Versione	Modifiche
30/03/2023	1.0	Prima stesura

Acronimi

CODICE	DESCRIZIONE
SGVP	Sistema di Gestione e Verifica Pass

Scopo del documento

Il documento ha lo scopo di descrivere le funzionalità del sistema di Gestione e verifica Pass (di seguito SGVP) , l'architettura generale e le procedure per l'installazione.

Contesto

Il sistema nasce dalla necessità di gestire tramite un portale i pass emessi dal Comune di Bari utilizzati per il riconoscimento:

- dei veicoli dell'autoparco
- degli operatori dello staff nominato in caso di eventi gestiti dal Comune di Bari

I Pass generati dal SGVP sono riprodotti su supporto cartaceo ed espongono un QR Code che contiene l'url sul dominio certificato:

verificapass.comune.bari.it

di una pagina ad accesso pubblico contenente le informazioni legate all'entità a cui è associato il Pass.

Architettura

SGVP è realizzato con le seguenti tecnologie:

- django framework versione 4.1.7 (framework python per lo sviluppo di portali web basato su pattern MVC)
- PostgreSQL versione 12
- Bootstrap Italia versione 2.0.9 (integrato nel pacchetto python django-design-theme:2.0.9)
- Altre librerie python per gestione dei PDF e miglioramento del rendering frontend.

Installazione

Per l'installazione seguire i passi sottostanti:

- Installazione dell'ambiente python richiesto dalla versione django (4.1.7) con versione ≥ 3 , per ulteriori dettagli fare riferimento alla guida ufficiale di python
- Individuare una cartella dove inserire i sorgenti del progetto (di seguito ROOT_PROJ) e posizionarsi all'interno con il prompt dei comandi
- Clonare il repository del progetto
- Creazione del virtual env di python in modo da isolare le dipendenze python di SGVP dai restanti progetti python installati localmente, nella cartella nominata **.venv** .

```
python -m venv .venv
```

- Attivazione della shell del virtualenv creato:
in caso di sistema operativo linux lanciare:
`source ../.venv/bin/activate`
- Installazione delle dipendenze di progetto:
`python -m pip install -r requirements.txt`
- Installazione del DBMS PostgreSQL (fare riferimento alla documentazione ufficiale)
- Creazione dell'utenza per l'accesso al DB es:
`CREATE ROLE sgvp NOSUPERUSER NOCREATEDB NOCREATEROLE NOINHERIT LOGIN
NOREPLICATION NOBYPASSRLS PASSWORD 'sgvp';`
- Creazione del database per SGVP e assegnazione dell'owner all'utente appena creato (es. sgvp)
- Creazione del file che contiene tutte le variabili di environment es. **prod.env** e inserire all'interno le variabili (l'esempio sottostante si riferisce al SO Linux):

```
DJANGO_SECRET_KEY=xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx  
DJANGO_DEBUG=False  
  
DATABASE_URL=postgres://userxxx:passwdxxx@ipdbms:port/dbsgvp  
ALLOWED_HOSTS=localhost,127.0.0.1  
  
#url con placeholder che verrà utilizzato per creare l'URL del pass da  
codificare nel QR code  
ORIGIN_URL_PASS_DETAIL=https://urlpubblica/pass/{badge_id}/detail  
CSRF_TRUSTED_ORIGINS=https://urlpubblica  
ENABLE_LDAP=True  
LDAP_AUTH_URL=ldap://hostldap:portldap  
LDAP_AUTH_SEARCH_BASE="DC=foo,DC=bar,DC=it"  
LDAP_AUTH_ACTIVE_DIRECTORY_DOMAIN=MYCOMUNE  
LDAP_AUTH_CONNECTION_USERNAME=userldap  
LDAP_AUTH_CONNECTION_PASSWORD="passwordldap"  
LDAP_FILTRO_RESTRIZIONE_UTENTI_AUTENTICABILI=  
  
export DJANGO_SECRET_KEY DJANGO_DEBUG DATABASE_URL ALLOWED_HOSTS  
ORIGIN_URL_PASS_DETAIL CSRF_TRUSTED_ORIGINS ENABLE_LDAP LDAP_AUTH_URL  
LDAP_AUTH_SEARCH_BASE LDAP_AUTH_ACTIVE_DIRECTORY_DOMAIN  
LDAP_AUTH_CONNECTION_USERNAME LDAP_AUTH_CONNECTION_PASSWORD  
LDAP_FILTRO_RESTRIZIONE_UTENTI_AUTENTICABILI
```

- Modificare opportunamente i valori delle variabili, il sistema permette opzionalmente (ENABLE_LDAP=True) di utilizzare LDAP per autenticare gli utenti
- Posizionarsi nella sottocartella **dja_badge_site** e lanciare il comando di applicazione delle migrazioni DB:
`python manage.py migrate`
- Creare l'utente superuser:
`python manage.py createsuperuser`
- Per avviare il server lanciare `python manage.py runserver`

Il portale è disponibile all'url <http://localhost:8000/>



Installazione in ambiente Linux con nginx

Per installare il portale come servizio di SO Linux e utilizzare un webserver nginx vanno seguiti i passi indicati nella guida all'url:

<https://www.digitalocean.com/community/tutorials/how-to-set-up-django-with-postgres-nginx-and-gunicorn-on-ubuntu-20-04>

Installazione di gunicorn e creazione socket e file di configurazione Gunicorn:

```
sudo nano /etc/systemd/system/djabadge.socket
```

```
djabadge.socket:
```

```
[Unit]
```

```
Description=djabadge gunicorn socket
```

```
[Socket]
```

```
ListenStream=/run/djabadge.sock
```

```
[Install]
```

```
WantedBy=sockets.target
```

```
sudo nano /etc/systemd/system/djabadge.service
```

```
[Unit]
```

```
Description=gunicorn daemon
```

```
Requires=djabadge.socket
```

```
After=network.target
```

```
[Service]
```

```
User=user
```

```
Group=www-data
```

```
WorkingDirectory=/path_to_directory_ROOT_PROJ
```

```
ExecStart=/path_to_directory_ROOT_PROJ/gunicorn_start
```

```
[Install]
```

```
WantedBy=multi-user.target
```

Lanciare l'installazione del service e socket:

```
sudo systemctl start djabadge.socket
```

```
sudo systemctl enable djabadge.socket
```

```
sudo systemctl start djabadge.service
```

```
sudo systemctl enable djabadge.service
```

Installare il server nginx e configurare il "site" per SGVP:

creare il file `/etc/nginx/sites-available/sgvp`

ed inserire il contenuto:

```
server {  
    listen 8088;  
    server_name verificapass.comune.bari.it;  
  
    location = /favicon.ico { access_log off; log_not_found off; }  
    location /static/ {  
        root /path_to_directory_ROOT_PROJ;  
    }  
    location / {  
        include proxy_params;  
        proxy_pass http://unix:/run/djabadge.sock;  
    }  
}
```

Il webserver nginx espone anche il path **/static** dove risiedono le risorse statiche (js,css, immagini etc..)

Manuale funzionale

Dopo aver installato e collaudato il sistema, l'accesso alla homepage si presenta come in figura:



Portale di gestione e verifica Pass emessi dal Comune di BARI

Il sistema permette di verificare l'autenticità dei Pass a partire dal QR code stampato sugli stessi, garantendo la visualizzazione dei contenuti pubblici e privati, in base alle autorizzazioni degli utenti utilizzatori.

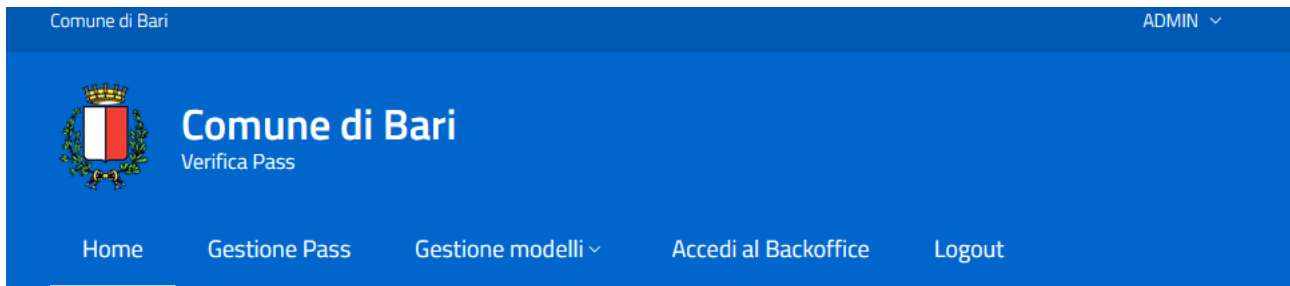
Per verificare il Pass, inquadra il QR code e verrai reindirizzato alla pagina di dettaglio dei dati del Pass, i dati privati saranno visibili solo agli utenti con i ruoli opportuni.

Per accedere alle funzionalità avanzate effettuare il [login](#)

Per qualsiasi dubbio e/o osservazione scrivere al seguente indirizzo privacy@comune.bari.it

In alto a destra è presente il tasto Accedi per effettuare il login.

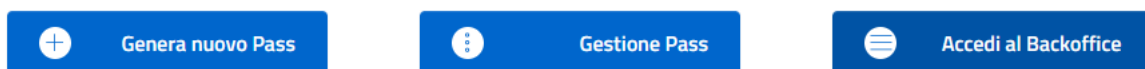
Ad accesso effettuato la home page mostra alcuni collegamenti alle funzioni principali e il menù principale si attiva a seconda del profilo dell'utente loggato.



Portale di gestione e verifica Pass emessi dal Comune di BARI

Il sistema permette di verificare l'autenticità dei Pass a partire dal QR code stampato sugli stessi, garantendo la visualizzazione dei contenuti pubblici e privati, in base alle autorizzazioni degli utenti utilizzatori.

Per verificare il Pass, inquadra il QR code e verrai rediretto alla pagina di dettaglio dei dati del Pass, i dati privati saranno visibili solo agli utenti con i ruoli opportuni.



Per qualsiasi dubbio e/o osservazione scrivere al seguente indirizzo privacy@comune.bari.it

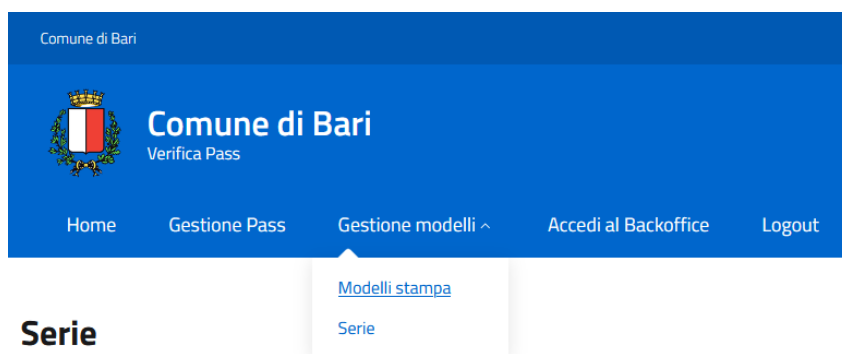
Prima della creazione dei pass vanno censite altre informazioni di seguito descritte.

Creazione del modello

Per la creazione dei Pass va definito il **modello** costituito da:

- File immagine utilizzato come sfondo nella riproduzione cartacea
- Coordinate di generazione e stampa del QR code
- Coordinate e font size per la stampa del codice autogenerato da sistema
- Nome univoco
- Set dei metadati associati al modello, per ogni metadato si definiscono:
 - Tipo metadato: (stringa, testo, data, file, immagine)
 - Nome
 - Eventuali coordinate in caso il metadato fosse marcato come stampabile
 - Attributi attivabili: stampabile, obbligatorio, privato


La funzione di creazione del modello è raggiungibile dal menu:



Per ogni modello la gestione dei metadati è raggiungibile dal tasto di azione con il simbolo di lista nella rispettiva riga del modello:

Comune di Bari

ADMIN ▾



Comune di Bari
Verifica Pass


[Home](#) [Gestione Pass](#) [Gestione modelli ▾](#) [Accedi al Backoffice](#) [Logout](#)

Elenco Modelli stampa

Ricerca

Nome modello contiene

Cerca

NOME MODELLO	ANTEPRIMA	
VEETURE AUTOPARCO		<div><div></div><div></div><div></div></div>

1

Torna alla home

Crea

Creazione della serie e sottoserie

Una volta definito il modello va censita la serie che rappresenta la prima classificazione dei Pass nel SGVP, ogni serie è definita da:

- Nome univoco
- Descrizione
- Periodo di validità
- Sigla (utilizzata per la prima parte del codice univoco universale **SSSSS-NNNNN**)
- Prog emissione (utilizzato per la seconda parte del codice univoco universale **SSSSS-NNNNN**)

Per ogni serie va definita almeno una sottoserie che rappresenta la subclassificazione dei Pass all'interno della serie. Ogni sottoserie è definita da:

- Serie
- Modello
- Periodo di validità

La creazione di serie è sottoserie è raggiungibile da menù Gestione Modelli come in figura:



Comune di Bari

Verifica Pass







[Home](#)[Gestione Pass](#)[Gestione modelli ^](#)[Accedi al Backoffice](#)[Logout](#)[Modelli stampa](#)[Serie](#)

Elenco Serie

Ricerca

Nome contiene

 Cerca

NOME	DESCRIZIONE	SIGLA	PROGR.EMISSIONE	DATA INIZIO VALIDITA	DATA FINE VALIDITA	
CAPODANNO TEST	CAPODANNO TEST	TST	1	09 Marzo 2023		<div><div></div><div></div><div></div></div>
AUTOVEICOLI AUTOPARCO	AUTOVEICOLI AUTOPARCO	ATP	51	08 Marzo 2023		<div><div></div><div></div><div></div></div>

La funzione di creazione dei Pass è raggiungibile dalla voce di menù Gestione Pass che punta ad una pagina con l'elenco dei Pass accessibili all'utente con i pulsanti nel footer per creazione e download del file pdf dei pass filtrati in elenco.



Elenco Pass

Ricerca

Sotto serie



Descrizione contiene

Codice



Cerca

TIPO PASS	CODICE	DESCRIZIONE	VALIDO	
AUTOVEICOLI AUTOPARCO	ATP-045	DS 340 YD	✓	
AUTOVEICOLI AUTOPARCO	ATP-028	DA 878 RF	✓	
AUTOVEICOLI AUTOPARCO	ATP-026	DA 178 RK	✓	
AUTOVEICOLI AUTOPARCO	ATP-001	CM 455 LY	✓	
AUTOVEICOLI AUTOPARCO	ATP-046	DS 341 YD	✓	

1

...

11



Torna alla home



Scarica Pass



Crea nuovo Pass

Per ogni pass è possibile modificare i dati e scaricare il pdf della riproduzione cartacea del pass.

Verifica del pass

La verifica del pass consiste nella scansione con smartphone del QR code presente sul pass e l'accesso alla conseguente URL riconosciuta. La pagina di verifica si presenta come in figura:

Comune di Bari

Accedi

Comune di Bari

Pass Valido

Dettaglio PASS

Codice	ATP-045
Descrizione	DS 340 YD
Tipo Pass	AUTOVEICOLI AUTOPARCO
Inizio validità	24 Marzo 2023
Fine validità	

Visualizza altri dati

Metadati:

TARGA	DS 340 YD
VEICOLO	

Comune di Bari

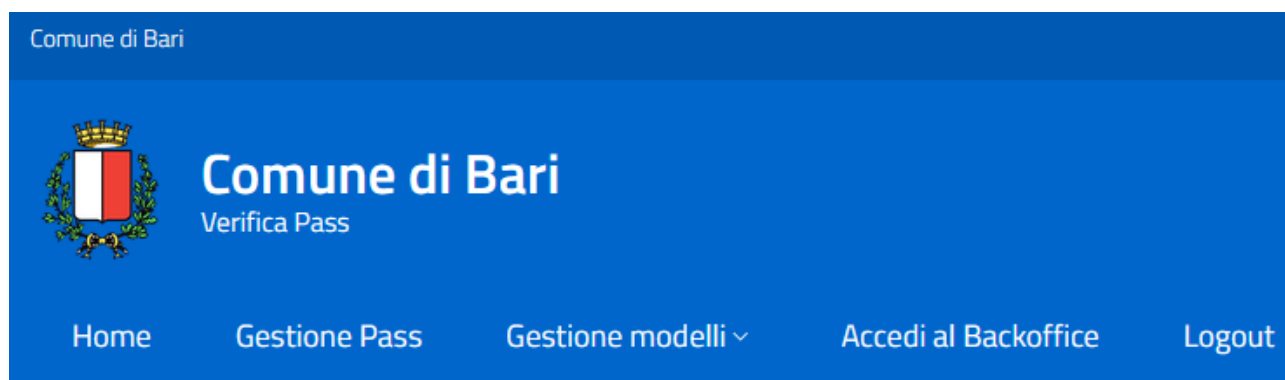
Il tasto visualizza altri dati viene mostrato in caso di utente non autenticato e permette di accedere ai metadati privati del Pass solo se l'utente ha il ruolo di Visualizzatore sulla serie del Pass. La validità è calcolata sui valori del periodo di validità.

Ruoli utente

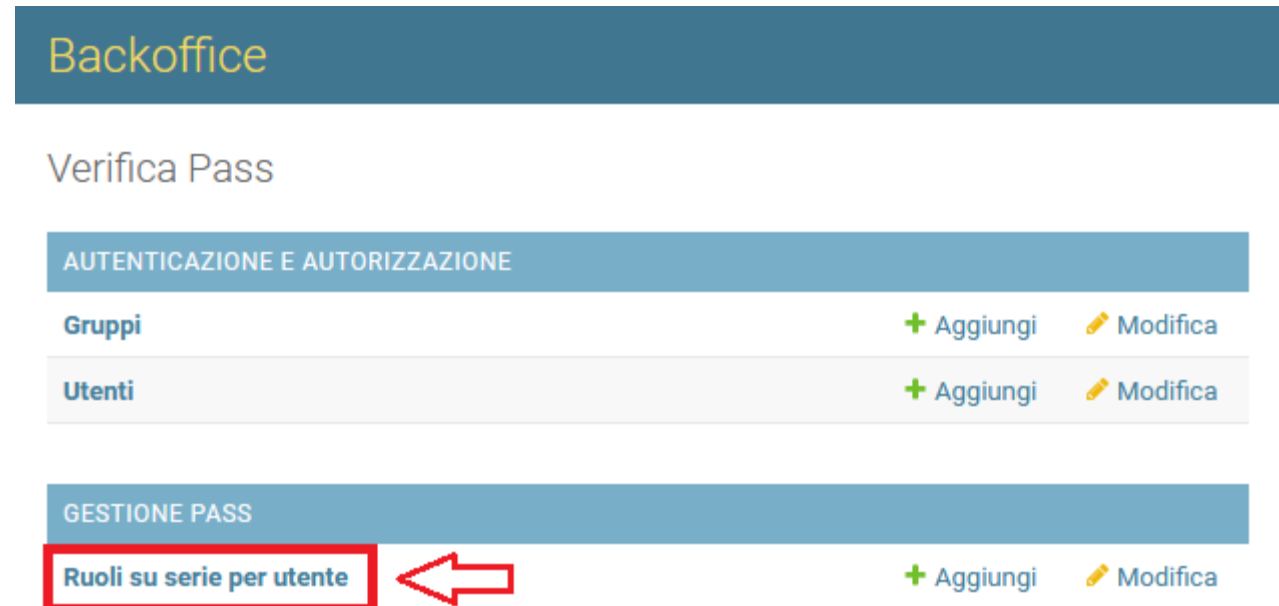
I ruoli previsti sono:

- Amministratore: nessuna restrizione sulle funzionalità
- Verificatore:
 - visualizza i dati “privati” del Pass che appartiene alle serie a cui è stato abilitato
- Gestore:
 - visualizza i dati “privati” del Pass che appartiene alle serie a cui è stato abilitato
 - crea e modifica i Pass sulle serie a cui è stato abilitato

Per l’assegnazione dei ruoli ad un utente censito va utilizzato il backoffice nativo di django raggiungibile dalla voce di menù Accedi al Backoffice, disponibile solo al ruolo Amministratore:



E navigando nella risorsa **ruoli su serie per utente**:



La creazione del ruolo prevede i campi obbligatori:

- ruolo (verificatore, gestore)
- utente
- periodo validità (data fine validità non obbligatoria)



Inizia a scrivere per filtrare...

AUTENTICAZIONE E AUTORIZZAZIONE

Gruppi + Aggiungi

Utenti + Aggiungi

GESTIONE PASS

Metadati pass + Aggiungi

Modelli metadato + Aggiungi

Modelli stampa pass + Aggiungi

Pass + Aggiungi

Ruoli su serie per utente + Aggiungi

Series + Aggiungi

Sotto series + Aggiungi

Aggiungi ruolo su serie per utente

Ruolo: Verificatore

User: Verificatore
Gestore

Serie:

Data inizio validita: Oggi

Data fine validita: Oggi

Salva e aggiungi un altro

Salva e continua le modifiche

SALVA