

Documentation Technique – BA380

Documentation Technique – Application BA380

1. Structure générale

L'application est une webapp Flask gérant deux bases SQLite : associations et bénévoles.

2. Technologies utilisées

- Flask (Python)
- HTML + Jinja2 + Bootstrap
- ReportLab, WeasyPrint
- gspread, googleapiclient
- Flask-Login
- Flask-Session (filesystem)
- .env pour config

3. Architecture

- app.py : logique principale
- export_to_sheets.py : envoi Google Sheets
- utils.py : fonctions communes (Drive, logs, DB)
- templates/ : HTML
- static/ : logo, CSS, etc.

4. Bases de données

Tables : associations, benevoles, field_groups, parametres, users

Champ `appli` = 'associations' ou 'benevoles'

5. Authentification

- Table users
- Rôle stocké en session : admin, gestionnaire, car, user

6. Sauvegarde Google Drive

- upload_database() via PyDrive2
- Suppression puis upload
- ID fichiers/dossiers dans .env

7. Export Google Sheets

- Fonctions : export_to_google_sheets_assos(), export_to_google_sheets_bene()
- Gère doublons via get_existing_spreadsheet_id()

8. PDF

- ReportLab pour fiches (fiche visite, annexe)
- WeasyPrint pour fiche_partenaire
- Images depuis static/images/

9. Scripts système

- admin_scripts déclenche .sh (backup, deploy, status...)
- Répertoire : /home/ndprz/scripts

10. Déploiement

- www.ba380.org (prod)
- ndprz.pythonanywhere.com (dev)
- Bases : ba380.sqlite, ba380dev.sqlite, *_test.sqlite
- Les fichiers .env permettent de configurer dynamiquement l'environnement
- Activation du mode test via session utilisateur (/set_test_user)