

Index projet - Documentation Centrale

Documents Références Rapides

Pour Démarrage Rapide

- **Nouveau sur le projet?** → Lire [README.md](#)
- **Besoin de setup?** → Section Installation [README.md#-installation](#)
- **Quels tests?** → Section Tests [README.md#-tests](#)

Pour Comprendre le Projet

- **Quelles sont les exigences?** → [CAHIER_DES_CHARGES.md](#)
 - Vue d'ensemble / Objectifs
 - Exigences fonctionnelles (☒ Complétées / ⌚ À venir)
 - Exigences non-fonctionnelles
 - Modèles de données
 - Critères d'acceptation

Pour Implémenter des Features

- **Comment ça marche techniquement?** → [PLAN_IMPLEMENTATION.md](#)
 - Architecture générale
 - Phase 1 & 2 (Complétées) - Détails d'implémentation
 - Phase 3 (À faire) - Graphiques, Recherche, PDF, Historique
 - Phase 4 (Production) - Déploiement, Sécurité, Maintenance
 - Procédures de test
 - Commandes utiles

Pour Suivre la Progression

- **Où en sommes-nous?** → [PROJECT_STATUS.md](#)
 - Résumé exécutif / KPIs
 - Détail par phase (% complété)
 - Métriques de qualité
 - Risques et mitigation
 - Prochaines étapes
 - Budget temps

Navigateur par Besoin

"Je dois ajouter une nouvelle feature"

1. Lire [CAHIER_DES_CHARGES.md](#) - Décrire en détail ce qui doit être fait
2. Consulter [PLAN_IMPLEMENTATION.md](#) - Voir comment implémenter
3. Vérifier [PROJECT_STATUS.md](#) - Voir dépendances/bloquants

4. Implémenter en suivant "Procédures de Test" du plan

"Je dois corriger un bug"

1. Reproduire en local (`python manage.py runserver`)
2. Identifier le composant (backend/frontend → voir architecture en [PLAN_IMPLEMENTATION.md](#))
3. Écrire un test qui reproduit le bug ([PROJECT_STATUS.md#métriques-de-qualité](#))
4. Corriger le code
5. Vérifier tous les tests `python manage.py test`

"Je dois déployer en production"

1. Consulter [PLAN_IMPLEMENTATION.md#6-phase-4-production](#)
2. Utiliser checklist déploiement: [PLAN_IMPLEMENTATION.md#8-checklist-déploiement](#)
3. Options PythonAnywhere/Heroku/VPS expliquées

"Je dois comprendre l'état du projet"

1. Lire [PROJECT_STATUS.md](#) - Vue globale
2. Détail par composant: [PROJECT_STATUS.md#7-status-par-composant](#)
3. Risques et mitigations: [PROJECT_STATUS.md#5-risques-et-mitigation](#)

Structure Fichiers & Références Direct

Documentation

.	
├─ README.md	← START HERE
├─ CAHIER_DES_CHARGES.md	← Spécifications
├─ PLAN_IMPLEMENTATION.md	← Techniques/Roadmap
├─ PROJECT_STATUS.md	← État actuel
├─ DOCUMENTATION_INDEX.md	← CE FICHER
└─ requirements.txt	← Dépendances Python

Code Source

backend/	
├─ manage.py	← Django CLI
├─ backend/	← Configurations Django
│ ├─ settings.py	
│ ├─ urls.py	
│ ├─ wsgi.py	
│ └─ asgi.py	
└─ notes/	
├─ models.py	← BD (Étudiant, UE, Note, etc.)
├─ views.py	← Logique métier + APIs
└─ urls.py	← Routing

```
|— admin.py          ← Admin Django
|— tests.py          ← Tests unitaires (12/12 ☒)
|— templates/notes/
|   |— base_adminlte.html ← Template master
|   |— home_adminlte.html ← Dashboard
|   |— tableau_notes_adminlte.html ← Notes table
|— static/js/
|   |— grades.js        ← Frontend controller (381 lines)
|— migrations/
|   |— 0001_initial.py
|   |— 0002_ue_note.py
```

Processus Typiques

Workflow Quotidien Développeur

```
# 1. Démarrer le serveur
cd backend
python manage.py runserver

# 2. Les tests au besoin
python manage.py test
coverage run --source='.' manage.py test # Aussi voir la couverture

# 3. Faire des changements
# ... Edit code ...
# (les templates auto-reload, les vues auto-reload)

# 4. Committer les changements
git add .
git commit -m "feat: descriptif court"

# 5. Push à GitHub (si public)
git push origin main
```

Workflow Avant Release

1. Lire PLAN_IMPLEMENTATION.md - Voir si phase est terminée
2. Vérifier PROJECT_STATUS.md - Tous les items ☒ ?
3. Exécuter tous les tests: `python manage.py test`
4. Code review: vérifier README pour clarté
5. Tag version: `git tag v0.3`
6. Faire release note
7. Committer + push

🕒 Chronologie Projet

Janvier 2026 → Phase 1 Foundation (20h) - ☒ DONE

- Setup Django, Models, Admin, Auth, Tests

Février 2026 (1-10) → Phase 2 Tableau Notes (25h) - ☒ DONE

- Dashboard, Filtres cascadants, APIs, Import/Export
- Tests 12/12 ☒

Février 2026 (11) → Documentation (3h) - ☒ DONE

- CAHIER_DES_CHARGES.md
- PLAN_IMPLEMENTATION.md
- PROJECT_STATUS.md

Février-Mars 2026 → Phase 3 UX Improvements (25-30h) - ⌚ PLANNING

- Graphiques Chart.js
- Recherche étudiant
- Validations avancées
- Export PDF
- Historique modifications

Mars-Avril 2026 → Phase 4 Production (15-20h) - 📋 PLANNING

- Hosting setup (PythonAnywhere/Heroku/VPS)
- PostgreSQL migration
- Security configuration
- Monitoring/Backups

🎓 Ressources d'Apprentissage

Language/Framework Docs

- [Django 4.2 Official Docs](#) - ☆ Référence le plus important
- [AdminLTE 4](#) - UI/UX components, theming
- [Bootstrap 5.3](#) - CSS framework

Libraries Utilisées

- [openpyxl](#) - Excel read/write
- [reportlab](#) - PDF generation (PDF export future)
- [Pillow](#) - Image processing

Tools Supplémentaires (Recommandés future)

- [Django Rest Framework](#) - API serialization
- [Chart.js](#) - Graphiques (Phase 3)
- [Webpack](#) - Asset bundling (optional)

☒ Checklist Accès au Projet

Quand tu commences:

- ☐ Cloné le repo: `git clone ...`
- ☐ Python 3.8+ installé: `python --version`
- ☐ venv créé: `python -m venv venv`
- ☐ Dépendances installées: `pip install -r requirements.txt`
- ☐ `cd backend` et migrations appliquées: `python manage.py migrate`
- ☐ Superuser créé: `python manage.py createsuperuser`
- ☐ Tests passent: `python manage.py test` (12/12 ☒)
- ☐ Serveur démarre: `python manage.py runserver`
- ☐ Tu peux accéder `http://127.0.0.1:8000/` ☒

Support et Contact

- Questions techniques? → Voir [PROJECT_STATUS.md#13-contact--support](#)
- Bug à reporter? → Incluez les détails + stack trace
- Améliorations suggérées? → Créez une issue GitHub

Vue Générale KPIs

Métrique	Status
Tests passants	<input checked="" type="checkbox"/> 12/12
Code complet Phase 2	<input checked="" type="checkbox"/> 100%
Documentation	<input checked="" type="checkbox"/> Complète
Production ready	<input type="radio"/> After Phase 3
Page load time	<input checked="" type="checkbox"/> < 2s

Document généré: 11 février 2026
Dernière mise à jour: 11 février 2026
Version: 1.0

Pour poser des questions ou clarifications, consulter [PROJECT_STATUS.md](#) section Contact.