

UNIVERSITÉ [NOM DE L'UNIVERSITÉ]

FACULTÉ D'INFORMATIQUE

PLATEFORME DE E-LEARNING AVANCÉE

Projet d'Architecture Logicielle & Programmation Java

Réalisé par : Alae Mamouni

N° Étudiant : [Votre numéro]

Niveau : Avancé

Encadré par : [Nom du professeur]

Date : Janvier 2026

Lien GitHub : https://github.com/alaemamouni963/Projet_E-Learning

Table des matières

1	Introduction et Contexte	3
1.1	Contexte du Projet	3
1.2	Problématique	3
1.3	Solution Proposée	3
2	Objectifs du Projet	4
2.1	Objectifs pédagogiques	4
2.2	Compétences développées	4
3	Architecture Logicielle	5
3.1	Architecture en Couches	5
3.2	Principes respectés	5
4	Diagrammes UML	6
4.1	Diagramme de classes	6
4.2	Relations clés	6
5	Design Patterns Implémentés	7
5.1	Singleton	7
5.2	Factory	7
5.3	Strategy	7
5.4	Observer	7
5.5	Decorator	7
6	Fonctionnalités	8
6.1	Gestion des utilisateurs	8
6.2	Cours et examens	8
7	Guide d'Installation	9
7.1	Prérequis	9
7.2	Installation	9
8	Conclusion	10

A Annexes	11
A.1 Bibliographie	11

Chapitre 1

Introduction et Contexte

1.1 Contexte du Projet

Ce projet a été réalisé dans le cadre du module *Architecture Logicielle Avancée*. Il consiste à concevoir et développer une plateforme complète de e-learning en Java, permettant la gestion des utilisateurs, des cours, des examens et des certifications.

1.2 Problématique

Comment concevoir une application Java robuste, maintenable et extensible capable de gérer un système complexe d'apprentissage en ligne avec plusieurs rôles et fonctionnalités avancées ?

1.3 Solution Proposée

La solution repose sur une architecture en couches, une séparation claire des responsabilités et l'utilisation de plusieurs design patterns reconnus.

Chapitre 2

Objectifs du Projet

2.1 Objectifs pédagogiques

- Maîtrise de la programmation orientée objet avancée
- Implémentation de design patterns
- Conception d'une architecture scalable
- Gestion de la persistance et de la sécurité

2.2 Compétences développées

- Java avancé
- Architecture logicielle
- Validation et sécurité
- Gestion des exceptions

Chapitre 3

Architecture Logicielle

3.1 Architecture en Couches

L'application est organisée en cinq couches principales :

- Présentation
- Service
- DAO
- Patterns
- Modèle

3.2 Principes respectés

- SOLID
- DRY
- KISS
- Faible couplage

Chapitre 4

Diagrammes UML

4.1 Diagramme de classes

Les diagrammes UML sont disponibles dans le dossier `/uml` du projet.

4.2 Relations clés

- Héritage : User → Student, Professor, Admin
- Composition : Course → Module → Lesson
- Association : Student ↔ Course

Chapitre 5

Design Patterns Implémentés

5.1 Singleton

```
public class Database {  
    private static Database instance;  
    private Database() {}  
    public static Database getInstance() {  
        if (instance == null) instance = new Database();  
        return instance;  
    }  
}
```

5.2 Factory

Centralise la création des contenus pédagogiques.

5.3 Strategy

Permet de changer l'algorithme de notation dynamiquement.

5.4 Observer

Notifie les utilisateurs lors des événements importants.

5.5 Decorator

Ajoute dynamiquement des fonctionnalités aux contenus.

Chapitre 6

Fonctionnalités

6.1 Gestion des utilisateurs

- Inscription et connexion sécurisées
- Trois rôles utilisateurs
- Gestion des sessions

6.2 Cours et examens

- Création de cours
- Progression des étudiants
- Examens et certifications

Chapitre 7

Guide d'Installation

7.1 Prérequis

Java JDK 11 ou supérieur.

7.2 Installation

```
git clone https://github.com/alaemamouni963/Projet_E-Learning.git
```

Chapitre 8

Conclusion

Ce projet a permis d'appliquer concrètement les concepts avancés de Java et d'architecture logicielle, tout en développant une application robuste et évolutive.

Annexe A

Annexes

A.1 Bibliographie

- Design Patterns – GoF
- Effective Java – Joshua Bloch
- Clean Code – Robert C. Martin