



Assistance Médicale d'Urgence – Description du Projet (Basée sur ta structure)

Assistance Médicale d'Urgence est une application web réalisée en **Java EE** suivant l'architecture **MVC (Model–View–Controller)**. Elle permet de gérer et coordonner les situations d'urgence médicale grâce à une interface web destinée aux administrateurs et aux équipes médicales.

Le projet est structuré autour de plusieurs modules : **interface web (JSP/CSS)**, **contrôleurs**, **couche métier**, **accès aux données (DAO)** et **gestion des ressources internes**.

Architecture du projet

Voici l'explication adaptée à ta structure :

◆ 1. `src/` – Code source principal

Contient toute la logique applicative :

a) `controller/`

- Classes Servlet (Java EE)
- Gèrent les requêtes HTTP (GET/POST)
- Assurent le lien entre la vue (JSP) et la couche métier
- Exemple typique : `UrgenceController`, `UserController`, etc.

b) `metier/`

- Contient la *couche métier*
- Regroupe les classes qui définissent la logique fonctionnelle :

- gestion des urgences
- gestion des utilisateurs
- gestion des équipes médicales
- Représentation des objets métiers (Urgence, Patient, Ambulance...)

c) dao/

- Couche d'accès aux données
- Classes responsables de :
 - Connexion à la base de données
 - CRUD (Create, Read, Update, Delete)
 - Requêtes SQL
- Suit le pattern DAO pour organiser la persistance des données.

d) META-INF/

- Fichiers de configuration (persistence.xml, etc.)
- Déclaration des ressources (connexion BD, ORM si utilisé)

◆ 2. WebContent/ – Interface Web & ressources

C'est la partie visible par l'utilisateur :

a) css/

- Fichiers de styles
- Mise en forme des pages JSP
- Gestion du thème visuel : couleurs, mise en page, boutons...

b) images/

- Logos, icônes, illustrations de l'application

c) view/

Dossier principal des pages JSP.

- Contient les formulaires, tableaux, dashboards...
- Sous-dossier **admin/** pour l'espace d'administration
Ex : gestion des utilisateurs, gestion des urgences, statistiques...

d) WEB-INF/

- Dossier protégé (non accessible directement via URL)
- Contient le web.xml (configuration des servlets)
- Le dossier lib/ contient les fichiers JAR nécessaires au fonctionnement :
 - JDBC
 - JSTL
 - Servlet API
 - etc.

◆ 3. build/

- Classes compilées
- Structure générée automatiquement par Eclipse
- Contient les .class de controller, dao, metier
- Le projet final s'exécute à partir de ces fichiers

🔗 Fonctionnalités principales de l'application

Selon la structure, voici ce que ton projet réalise typiquement :

- Gestion des utilisateurs (admin, équipes médicales)
- Création et suivi des urgences médicales
- Affectation des intervenants
- Tableau de bord administratif
- Gestion des données via DAO et base SQL
- Interface web dynamique (JSP / Servlets)