



# From Software

Ensemble des incidents

Alves Fernandes  
INTERWEB



# From Software



## Sommaire

### Table des matières

<b>Sommaire .....</b>	<b>1</b>
<b>I- Les erreurs de codes. ....</b>	<b>2</b>
1. Erreur de suppression .....	2
2. Erreur de nomination .....	3
3. Erreur Fichier en SQL. ....	4

# From Software

## I- Les erreurs de codes.

### 1. Erreur de suppression

❗ Erreur fatale : PDOException non détectée : SQLSTATE[23000] : Violation de la contrainte d'intégrité : 1451 Impossible de supprimer ou de mettre à jour une ligne parent : une contrainte de clé étrangère échoue ('recrute\_db'. 'candidature', CONSTRAINT 'Candidature\_Annonce\_FK' FOREIGN KEY ('idAnnonce') REFERENCES 'annonce' ('idAnnonce')) in C : \wamp64\www\Site\_Epreuve\_E5(new)\modele\annonceModel.php on line 32

❗ PDOException : SQLSTATE[23000] : Violation de la contrainte d'intégrité : 1451 Impossible de supprimer ou de mettre à jour une ligne parente : une contrainte de clé étrangère échoue ('recrute\_db'. 'candidature', CONSTRAINT 'Candidature\_Annonce\_FK' FOREIGN KEY ('idAnnonce') REFERENCES 'annonce' ('idAnnonce')) in C : \wamp64\www\Site\_Epreuve\_E5(new)\modele\annonceModel.php on line 32

Pile d'appels

#	Heure	Mémoire	Fonction	Emplacement
1	0.0001		452240 {main}()	...annonceControleur.php :0
2	0.0022		585320 AnnonceController->delete()	...annonceControleur.php :17
3	0.0026		589560 Annonce->deleteAnn( \$idA = '13' )	...annonceControleur.php :60
4	0.0026		590256 execute( \$params = [0 => '13'] )	...annonceModel.php :32

- Quand on voulait supprimer l'offre cette erreur apparaissait.
- La contrainte de clé étrangère empêche cette opération pour préserver l'intégrité des données.
- Nous avons réussi à régler ce problème en faisant en sorte que lorsqu'on veut supprimer une offre alors automatiquement les candidatures liées seront supprimé.

```
1 reference | 0 overrides
function deleteAnn($idA): bool
{
    // Vérifier si l'annonce existe
    $req = $this->bdd->prepare("SELECT COUNT(*) FROM Annonce WHERE idAnnonce = ?");
    try {
        // Démarrer une transaction
        $this->bdd->beginTransaction();

        // Supprimer les candidatures associées à l'annonce
        $req = $this->bdd->prepare("DELETE FROM candidature WHERE idAnnonce = ?");
        $req->execute([$idA]);

        // Supprimer l'annonce
        $req = $this->bdd->prepare("DELETE FROM Annonce WHERE idAnnonce = ?");
        $req->execute([$idA]);

        // Valider la transaction
        $this->bdd->commit();
        return true;
    } catch (PDOException $e) {
        // Annuler la transaction en cas d'erreur
        $this->bdd->rollBack();
        throw new Exception(message: "Erreur lors de la suppression : " . $e->getMessage());
    }
}
```

# From Software

## 2. Erreur de nomination

Au cours du projet nous avons eu différents comme la dénomination.

Par exemple :

- Mauvais nom de clé dans le tableau `\$\_SESSION` dans le fichier annonce controleur.

Une erreur de type `Undefined index: details` sera levée, car la clé `details` n'existe pas dans `\$\_SESSION`.

Modifiez légèrement le nom de la clé utilisée pour accéder à l'ID de l'annonce, ce qui provoquera une erreur.

```
public function updateAnn(){
    session_start();
    $this->annonce-
>updateAnn($_POST['titre'],$_POST['adresse'],$_POST['typeCon'],$_POST['description'],
$_POST['salaire'],$_POST['duree'],$_SESSION['details']['idAnnonce']);
    $_SESSION['detail']['titre']=$_POST['titre'];
    $_SESSION['detail']['adresse']=$_POST['adresse'];
    $_SESSION['detail']['typeCon']=$_POST['typeCon'];
    $_SESSION['detail']['description']=$_POST['description'];
    $_SESSION['detail']['salaire']=$_POST['salaire'];
    $_SESSION['detail']['duree']=$_POST['duree'];
    header('Location: '.$_SESSION['location'].'/index.php?page=8');
    exit;
}
```

- Pour cette erreur, il faut juste mettre la bonne clé qui est « detail » sans le (s).

# From Software

## 3. Erreur Fichier en SQL.

### Erreur : Utilisation d'un type inapproprié pour stocker des fichiers

Au cours du projet nous voulions que le candidat puisse mettre leur cv quand il se candidate pour un poste. Nous avons utilisé VARCHAR ou TEXT pour stocker des fichiers binaires (comme des PDF), cela causé des problèmes, car ces types ne sont pas conçus pour gérer de grandes quantités de données binaires.

Pour résoudre ce problèmes nous avons utilisé LONGBLOB pour le cv.

```
CREATE TABLE Candidature (  
    idCandidature INT AUTO_INCREMENT NOT NULL,  
    cv LONGBLOB NOT NULL, -- Type adapté pour stocker des  
fichiers binaires  
    motivation LONGBLOB NULL,  
    extraDoc LONGBLOB NULL,  
    statut VARCHAR(50) NOT NULL,  
    idCandidat INT NOT NULL,  
    idAnnonce INT NOT NULL,  
    dateCand VARCHAR(50) NOT NULL,  
    CONSTRAINT Candidature_PK PRIMARY KEY (idCandidature)  
) ENGINE=InnoDB;
```