**Projet GSB C#**



**Sommaire**

1. Présentation du Projet
2. Le MCD
3. Base de données
4. Code Source
   1. Les classes
      1. Responsable
      2. Technicien
      3. User
      4. Matériel
      5. Incident
   2. La classe BD
5. Les interfaces
6. Le cahier de recette
7. Manuel utilisation

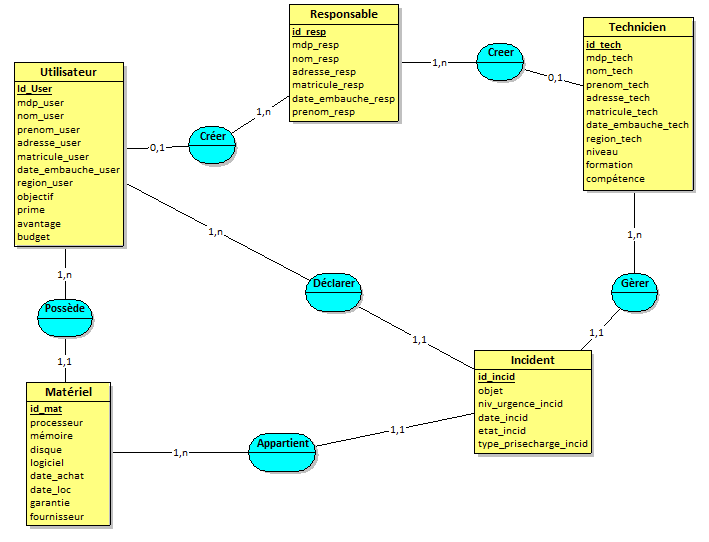
**Présentation du Projet**

Le but de ce projet consiste à créé une application pour des Responsables, des techniciens et des utilisateurs dans lequel ils pourront créer des profils pour le personnel, enregistrer du matériels, ajouter des incidents, l’affichage d’incidents et de matériels et bien plus… comme des statistiques.

Il nous faut donc développé ce programme en créant des classes pour chaque type d’utilisateurs, des classes pour matériel et incident et tout ça en interaction avec une base de donnée afin de stocker le contenu et pouvoir faire les requêtes nécessaires lorsqu’on en aura besoin.

**Le MCD**

Création du Modèle Conceptuel de Donnée :



**Base de données**

Image de la base de donnée représentant les classes de l’applications.

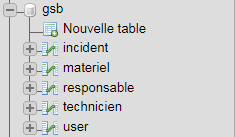


Table Incident :



Table Matériel :



Table Utilisateur :



Table Technicien :



Table Responsable :



**Code source**

Les classes suivantes contiennent toute les variables nécessaires, un constructeur et les getters setters.

**Classe Responsable :** responsable.cs

namespace Projet\_GSB

{

class responsable

{

private String mdpResp;

private String nomResp;

private String prenomResp;

private String adresseResp;

private String matriculeResp;

private String dateResp;

public responsable(String pMdpResp, String pNomResp, String pPrenomResp, String pAdresseResp, String pMatriculeResp, String pDateResp)

{

mdpResp = pMdpResp;

nomResp = pNomResp;

prenomResp = pPrenomResp;

adresseResp = pAdresseResp;

matriculeResp = pMatriculeResp;

dateResp = pDateResp;

}

//GETTER SETTER

public string MdpResp { get => mdpResp; set => mdpResp = value; }

public string NomResp { get => nomResp; set => nomResp = value; }

public string PrenomResp { get => prenomResp; set => prenomResp = value; }

public string AdresseResp { get => adresseResp; set => adresseResp = value; }

public string MatriculeResp { get => matriculeResp; set => matriculeResp = value; }

public string DateResp { get => dateResp; set => dateResp = value; }

}

}

**Classe Technicien :** technicien.cs

namespace Projet\_GSB

{

class technicien

{

private String mdpTech;

private String nomTech;

private String prenomTech;

private String adresseTech;

private String matriculeTech;

private String dateTech;

private String regionTech;

private int nivInter;

private String formation;

private String competence;

private int idResp;

public technicien(String pMdpTech, String pNomTech,String pPrenomTech,String pAdresseTech,String pMatriculeTech,String pDateTech,String pRegionTech,int pNivInter,String pFormation,String pCompetence, int pIdResp)

{

mdpTech = pMdpTech;

nomTech = pNomTech;

prenomTech = pPrenomTech;

adresseTech = pAdresseTech;

matriculeTech = pMatriculeTech;

dateTech = pDateTech;

regionTech = pRegionTech;

nivInter = pNivInter;

formation = pFormation;

competence = pCompetence;

idResp = pIdResp;

}

//GETTER SETTER

public string MdpTech { get => mdpTech; set => mdpTech = value; }

public string NomTech { get => nomTech; set => nomTech = value; }

public string PrenomTech { get => prenomTech; set => prenomTech = value; }

public string AdresseTech { get => adresseTech; set => adresseTech = value; }

public string MatriculeTech { get => matriculeTech; set => matriculeTech = value; }

public string DateTech { get => dateTech; set => dateTech = value; }

public string RegionTech { get => regionTech; set => regionTech = value; }

public int NivInter { get => nivInter; set => nivInter = value; }

public string Formation { get => formation; set => formation = value; }

public string Competence { get => competence; set => competence = value; }

public int IdResp { get => idResp; set => idResp = value; }

}

**Classe Utilisateur :** user.cs

namespace Projet\_GSB

{

class user

{

private String mdpUser;

private String nomUser;

private String prenomUser;

private String adresseUser;

private String matriculeUser;

private String dateUser;

private String regionUser;

private String objectif;

private float prime;

private String avantage;

private float budget;

private int idResp;

public user(String pMdpUser,String pNomUser,String pPrenomUser,String pAdresseUser,String pMatriculeUser,String pDateUser,String pRegionUser,String pObjectif,int pPrime,String pAvantage,int pBudget, int pIdResp)

{

mdpUser = pMdpUser;

nomUser = pNomUser;

prenomUser = pPrenomUser;

adresseUser = pAdresseUser;

matriculeUser = pMatriculeUser;

dateUser = pDateUser;

regionUser = pRegionUser;

objectif = pObjectif;

prime = pPrime;

avantage = pAvantage;

budget = pBudget;

idResp = pIdResp;

//dateTimePicker1.Value.ToShortDateString();

}

//GETTER SETTER

public string MdpUser { get => mdpUser; set => mdpUser = value; }

public string NomUser { get => nomUser; set => nomUser = value; }

public string PrenomUser { get => prenomUser; set => prenomUser = value; }

public string AdresseUser { get => adresseUser; set => adresseUser = value; }

public string MatriculeUser { get => matriculeUser; set => matriculeUser = value; }

public string DateUser { get => dateUser; set => dateUser = value; }

public string RegionUser { get => regionUser; set => regionUser = value; }

public string Objectif { get => objectif; set => objectif = value; }

public float Prime { get => prime; set => prime = value; }

public string Avantage { get => avantage; set => avantage = value; }

public float Budget { get => budget; set => budget = value; }

public int IdResp { get => idResp; set => idResp = value; }

}

}

**Classe Materiel :** materiel.cs

namespace Projet\_GSB

{

class materiel

{

private String processeur;

private int memoire;

private String disque;

private String logiciel;

private String dateAchatLoc;

private int garantie;

private String fournisseur;

private int idTech;

public materiel(String pProcesseur, int pMmemoire, String pDisque, String pLogiciel, String pDateAchatLoc, int pGarantie, String pFournisseur, int pIdTech)

{

processeur = pProcesseur;

memoire = pMmemoire;

disque = pDisque;

logiciel = pLogiciel;

dateAchatLoc = pDateAchatLoc;

garantie = pGarantie;

fournisseur = pFournisseur;

idTech = pIdTech;

}

public string Processeur { get => processeur; set => processeur = value; }

public int Memoire { get => memoire; set => memoire = value; }

public string Disque { get => disque; set => disque = value; }

public string Logiciel { get => logiciel; set => logiciel = value; }

public string DateAchatLoc { get => dateAchatLoc; set => dateAchatLoc = value; }

public int Garantie { get => garantie; set => garantie = value; }

public string Fournisseur { get => fournisseur; set => fournisseur = value; }

public int IdTech { get => idTech; set => idTech = value; }

}

}

**Classe Incident :** incident.cs

namespace Projet\_GSB

{

class incident

{

private String objet;

private int nivUrgence;

private String dateIncid;

private int etatIncid;

private String typePriseCharge;

private int idMat;

private int idTech;

private int idUser;

public incident(String pObjet, int pNivUrgence, String pDateIncid, int pEtatIncid, String pTypePriseCharge, int pIdMat, int pIdTech, int pIdUser)

{

objet = pObjet;

nivUrgence = pNivUrgence;

dateIncid = pDateIncid;

etatIncid = pEtatIncid;

typePriseCharge = pTypePriseCharge;

idMat = pIdMat;

idTech = pIdTech;

idUser = pIdUser;

}

//GETTER SETTER

public string Objet { get => objet; set => objet = value; }

public int NivUrgence { get => nivUrgence; set => nivUrgence = value; }

public string DateIncid { get => dateIncid; set => dateIncid = value; }

public int EtatIncid { get => etatIncid; set => etatIncid = value; }

public string TypePriseCharge { get => typePriseCharge; set => typePriseCharge = value; }

public int IdMat { get => idMat; set => idMat = value; }

public int IdTech { get => idTech; set => idTech = value; }

public int IdUser { get => idUser; set => idUser = value; }

}

}

**La Classe BD**

Entête seulement :

static string connexion = "Server=127.0.0.1;Database=gsb;Uid=root;Password=;";

//Connexion Responsable

public static int connexionResp(string pMatricule, string pMdp)

//Connexion Utilisateur

public static int connexionUser(string pMatricule, string pMdp)

//Connexion Technicien

public static int connexionTech(string pMatricule, string pMdp)

//Ajoute User

public static void ajouteUser(user unUser)

//Ajoute Technicien

public static void ajouteTech(technicien unTech)

//Modification Utilisateur

public static void modifUser(String pMatricule, String pAdresse, String pRegion)

//Modification Technicien

public static void modifTech(String pMatricule, String pAdresse, String pRegion)

//Supprime User

public static void supprUser(String pMatricule)

//Supprime Yechnicien

public static void supprTech(String pMatricule)

//Ajout Matériel

public static void ajoutMateriel(materiel unMat)

//Supprime Materiel

public static void supprMat(String pId)

//Ajoute incident

public static void ajouteIncid(incident unIncid)

//Consulte incident

public static String consulteIncid(String pId)

//Affiche incident

public static List<incident> afficheIncidCollection()

//Changement Etat

public static void etat1(int pId, int idTech)

public static void etat2(int pId, int idTech)

//STATISTIQUE

//Compte nombre d'incident

public static int nbIncid()

public static int nbEnCharge()

public static int nbResolu()

public static int nbResoluTech()

public static int nbChargeTech()

**Code du main Form(Connexion)**

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void btnConnexion\_Click(object sender, EventArgs e)

{

if(radioUser.Checked)

{

int id = BD.connexionUser(TBmatricule.Text, TBmdp.Text);

if (id != 0)

{

Form2 formUsers = new Form2(id);

formUsers.ShowDialog();

this.Hide();

}

else

{

MessageBox.Show("Erreur MDP ou Matricule");

}

}

else if(radioTech.Checked)

{

int id = BD.connexionTech(TBmatricule.Text, TBmdp.Text);

if (id != 0)

{

Form3 formTech = new Form3(id);

formTech.ShowDialog();

this.Hide();

}

else

{

MessageBox.Show("Erreur MDP ou Matricule");

}

}

else if(radioResp.Checked)

{

int id = BD.connexionResp(TBmatricule.Text, TBmdp.Text);

if(id != 0)

{

Form4 formResp = new Form4(id);

formResp.ShowDialog();

this.Hide();

}

else

{

MessageBox.Show("Erreur MDP ou Matricule");

}

}}}

**Code du main Form(Utilisateur)**

public partial class Form2 : Form

{

int id;

public Form2(int pId)

{

InitializeComponent();

id = pId;

}

private void btnDeclarer\_Click(object sender, EventArgs e)

{

incident unIncid = new incident(TBprobleme.Text, Convert.ToInt32(numUrgence.Value), Convert.ToString(DateTime.Now), Convert.ToInt32(numEtat.Value), "Reçu", Convert.ToInt32(TBposte.Text),0, id);

BD.ajouteIncid(unIncid);

}

private void btnAffIncident\_Click(object sender, EventArgs e)

{

labelIncid.Text = BD.consulteIncid(TBposte2.Text);

}

}

**Code du main Form(Technicien)**

public partial class Form3 : Form

{

int id;

public Form3(int pId)

{

InitializeComponent();

id = pId;

}

private void btnAdd\_Click(object sender, EventArgs e)

{

materiel unMat = new materiel(TBproco.Text, Convert.ToInt32(TBmemoire.Text), TBdisque.Text, TBlogiciel.Text, Convert.ToString(DateTime.Now) , Convert.ToInt32(numGarantie.Value), TBfournisseur.Text, id);

BD.ajoutMateriel(unMat);

MessageBox.Show("Ajout Réussi");

}

private void button1\_Click(object sender, EventArgs e)

{

MessageBox.Show(BD.consulteMat(TBid.Text));

}

private void button2\_Click(object sender, EventArgs e)

{

BD.supprMat(TBdelMat.Text);

MessageBox.Show("Suppression Réussi !");

}

private void button3\_Click(object sender, EventArgs e)

{

List<incident> lesIncidents = BD.afficheIncidCollection();

int id = 0;

foreach(incident unIncident in lesIncidents)

{

++id;

listBoxIncid.Items.Add("id : " + id + "Etat : " + unIncident.TypePriseCharge + " " + " Niv D'urgence : " + unIncident.NivUrgence + " " + " Objet : " + unIncident.Objet);

}

}

private void button4\_Click(object sender, EventArgs e)

{

BD.etat1(Convert.ToInt32(numIncid.Value), id);

}

private void button5\_Click(object sender, EventArgs e)

{

BD.etat2(Convert.ToInt32(numIncid.Value), id);

}

**Code du main Form(Responsable)**

public partial class Form4 : Form

{

int id;

public Form4(int pId)

{

InitializeComponent();

id = pId;

}

private void listBox1\_SelectedIndexChanged(object sender, EventArgs e)

{

}

private void button1\_Click(object sender, EventArgs e)

{

//MessageBox.Show(Convert.ToString(id));

if(radioUser.Checked)

{

user unUser = new user(TBmdp.Text, TBnom.Text, TBprenom.Text, TBadresse.Text, TBmatricule.Text, dateTimePicker1.Value.ToString(), TBregion.Text, TBobjectif.Text, Convert.ToInt32(TBprime.Text), TBavantage.Text, Convert.ToInt32(TBbudget.Text), id);

BD.ajouteUser(unUser);

MessageBox.Show("Ajout Réussi !");

}

else if(radioTech.Checked)

{

technicien unTech = new technicien(TBmdp.Text, TBnom.Text, TBprenom.Text, TBadresse.Text, TBmatricule.Text, dateTimePicker1.Value.ToString(), TBregion.Text, Convert.ToInt32(TBniv.Text), TBformation.Text, TBcompetence.Text, id);

BD.ajouteTech(unTech);

MessageBox.Show("Ajout Réussi !");

}

else

{

MessageBox.Show("Veuillez choisir le type d'ajout !");

}

}

private void Form4\_Load(object sender, EventArgs e)

{

}

private void button3\_Click(object sender, EventArgs e)

{

if(rdModifUser.Checked)

{

BD.modifUser(TBmodifMatricule.Text, TBmodifAdresse.Text, TBmodifRegion.Text);

MessageBox.Show("Modification Réussi !");

}

else if(rdModifTech.Checked)

{

BD.modifTech(TBmodifMatricule.Text, TBmodifAdresse.Text, TBmodifRegion.Text);

MessageBox.Show("Modification Réussi !");

}

else

{

MessageBox.Show("Veuillez choisir le type que vous voulez modifier !");

}

}

private void radioButton1\_CheckedChanged(object sender, EventArgs e)

{

}

private void button2\_Click(object sender, EventArgs e)

{

if(rdDelUser.Checked)

{

BD.supprUser(TBdelMatricule.Text);

MessageBox.Show("Suppression Réussi !");

}

else if(rdDelTech.Checked)

{

BD.supprTech(TBdelMatricule.Text);

MessageBox.Show("Suppression Réussi !");

}

else

{

MessageBox.Show("Veuillez choisir le type de suppression !");

}

}

private void button4\_Click(object sender, EventArgs e)

{

labelIncidResolu.Text = Convert.ToString(BD.nbResolu());

labelIncidCharge.Text = Convert.ToString(BD.nbEnCharge());

labelNbIncid.Text = Convert.ToString(BD.nbIncid());

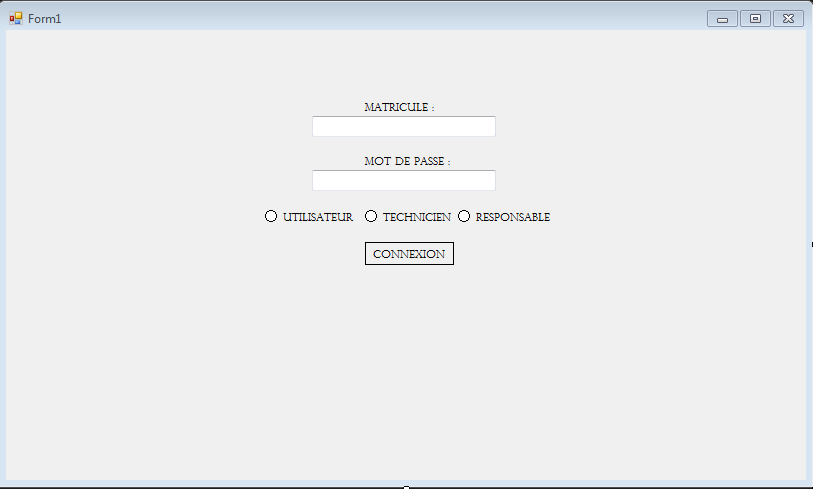
labelNbTechReso.Text = Convert.ToString(BD.nbResoluTech());

labelNbTechCharge.Text = Convert.ToString(BD.nbChargeTech());

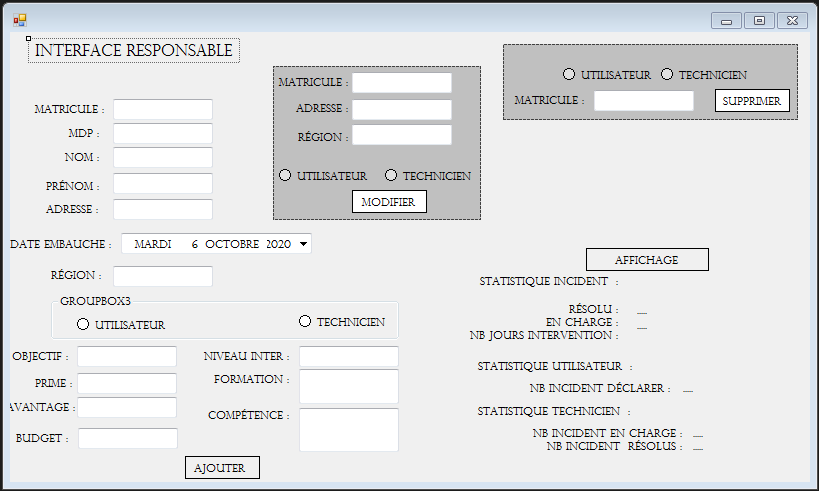
}

**Les Interfaces**

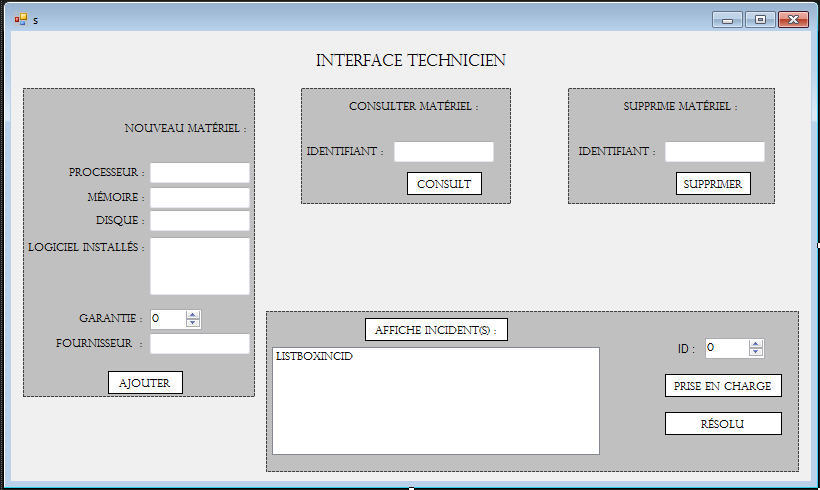
Interface Connexion :



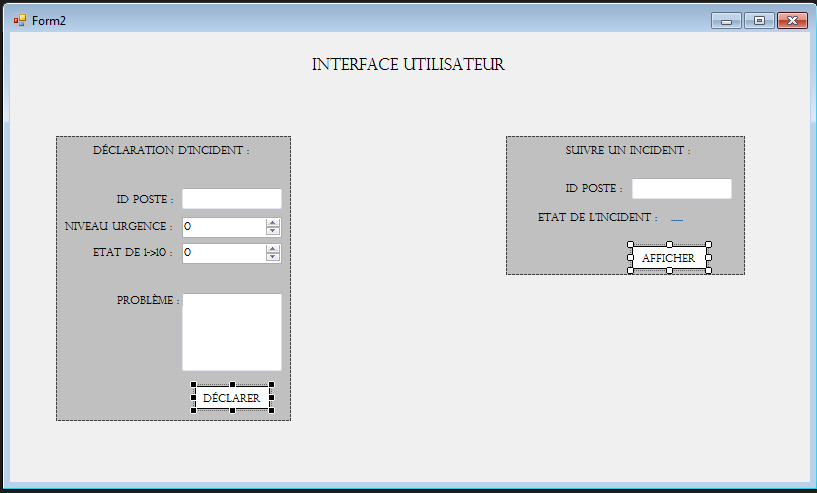
Interface Responsable :



Interface Technicien :



Interface Utilisateur :



**Le cahier de recette**

« Fourni en annexe dans le dossier. »

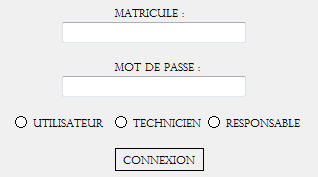
**Manuel D’utilisation**

**Pour l’inferface Connexion :**

* Saisir votre MATRICULE et votre Mot de Passe puis sélectionné votre type en fonction de votre rôle, Responsable, Technicien ou Utilisateur.

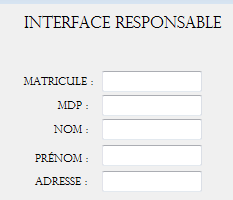
Puis un clic sur le bouton « Connexion » permettra l’accès à l’interface voulu.

* Si une erreur survient un Message apparaitra comme quoi le MATRICULE ou le MOT DE PASSE est érroné.



**Pour l’interface Responsable :**

* Pour la **création** d’un Utilisateur/Technicien le Responsable doit renseigner les champs suivant qui sont commun aux deux :

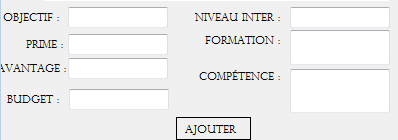




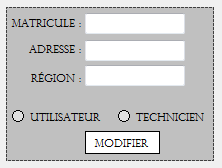
* Il doit ensuite choisir entre les 2 options disponibles :



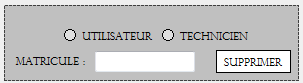
* Une fois le choix effectué il ne lui reste plus qu’à remplir les champs en dessous de l’option selectionné et de faire un clic sur le bouton « Ajouter »



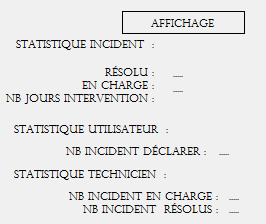
* Pour la **Modification** d’un Utilisateur/Technicien, le responsable doit saisir le MATRICULE de la personne à modifié et saisir la nouvelle adresse/région, il doit ensuite sélectionné si c’est un Utilisateur ou Technicien (en rapport avec son MATRICULE) et faire un clic sur « Modifier » pour effectué les modifications.



* Pour la **Suppression** d’un Utilisateur/Technicien le Responsable doit sélectionné le type et saisir le MATRICULE, puis un clic sur « Supprimer ».

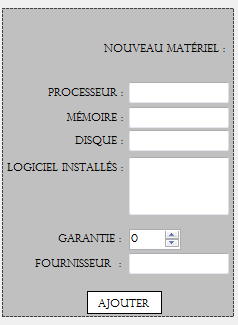


* Pour **l’Affichage** des statistiques une simple clic sur « Affichage » permettra l’affichage des données dites ci-dessous :

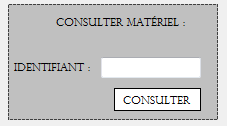


**Pour l’inferface Technicien :**

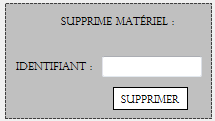
* Pour **l’Ajout** d’un materiel il suffit au technicien de saisir toute les informations nécessaires dans les camps ci-dessous :
* Un clic sur « Ajouter » permettra l’ajout dans la BD.

****

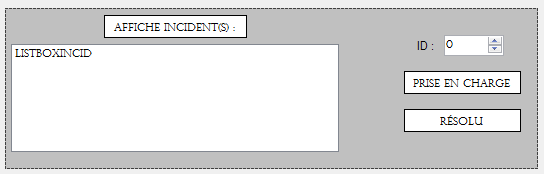
* Pour **Consulter** un matériel, il faudra remplir le champ de l’Identifiant du materiel voulu et faire un clic sur « Consulter »
* Un Message type PopUp apparaitra et affichera toute les informations du materiel saisi.



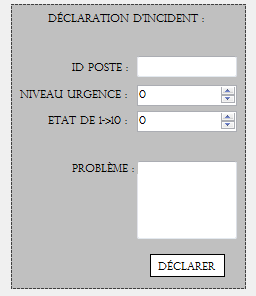
* Pour **Supprimer** un matériel le technicien devra saisir l’identifiant du matériel et faire un clic sur « Supprimer »



* Pour l’**Affichage** des incidents, un clic sur « Affiche Incident(s) » permettra l’affichage de ceux-ci, il faudra ensuite saisir l’id de l’incident souhaitez et faire un clic sur « Prise en charge » ou « Résolu » pour changer l’état de l’incident.



**Pour l’inferface Utilisateur :**

* L’utilisateur peut **Déclarer** un incident il devra remplir les données souhaitez lors de la déclaration et faire un clic sur « Déclarer »
* ****
* L’utilisateur peut aussi **Suivre** un incident, il lui suffit simplement de saisir l’id du poste souhaitez un clic sur « Afficher » et l’état de celui-ci lui sera fourni.

