



# PROJET

## Gestion des Tickets Incidents

**PPE 3 : Contexte laboratoire GSB**

**Option SLAM**

### **Activité à réaliser :**

Vous êtes chargé de développer une application **C#** à destination des visiteurs médicaux, afin qu'ils puissent gérer des tickets d'incidents.

- Rédaction du modèle Entités Associations (**MCD**) de la base de données.
- Création de la base de données sous **MySQL**.
- Création des classes nécessaires au traitement.
- Réalisation de l'application **C#** qui provoquera la mise à jour de la base de données.
- Rédaction de la documentation technique de l'application contenant le diagramme des **classes** que vous aurez développées.
- Rédaction d'un guide d'utilisation de cette application à destination des visiteurs médicaux.
- Prévoir un formulaire afin de récolter le consentement de l'utilisateur (clair et explicite) pour faire en sorte que votre application soit en conformité avec le **RGPD**.

**Votre travail devra prendre la forme d'un document écrit, relié et paginé**

**La présentation fera l'objet d'un oral de 15 minutes maximum, et chaque membre du groupe sera ensuite interrogé sur une activité choisie par le jury.**

**Début de la présentation des Travaux : courant Novembre 2018**

# Projet Assistance

## Gestion des tickets incidents

### I - Objectif

La mise en place des équipements et des applications pour 450 visiteurs chez le Laboratoire GSB va entraîner une inévitable demande d'assistance de la part du personnel.

La DSI désire mettre en place une solution de gestion des incidents permettant :

- de prendre en compte une demande (création d'un ticket d'incident),
- de l'affecter à un technicien,
- d'enregistrer la solution apportée (création d'une base de connaissance),
- d'informer l'utilisateur de la prise en charge de sa demande.

#### **1. Utilisateurs concernés**

Liste des utilisateurs visés :

- Responsables - Délégués
- Visiteurs
- Administrateur
- Techniciens

#### **2. Proposition d'une application développée en C#**

L'application intitulée *Gestion des tickets incidents* sera développée en *Visual C#*, la base de données sera gérée avec *MySQL ou SQL Server*.

##### **a. Codage**

Le codage de l'application doit respecter les normes de codage du langage C#.

##### **b. Environnement**

Le développement sera effectué à l'aide de Visual Studio C#, avec une base de données *MySQL ou SQL Server*.

##### **c. Graphisme et ergonomie**

L'allure générale de l'application se caractérisera par :

- Un respect de la charte graphique de GSB (logo, typographie).
- Les couleurs dominantes seront les suivantes :  
Gris et bleu.

## II - Spécifications fonctionnelles et techniques

### 3. Déroulé fonctionnel

La gestion de tickets incidents consiste à gérer les incidents via « l'application *Gestion des Tickets Incidents* ».

Elle répertorie les demandes de dépannage et d'intervention faites par les utilisateurs du Laboratoire *Galaxy Swiss Bourdin* sous forme de ticket.

Ces tickets sont affectés aux techniciens qui en auront la charge jusqu'à la clôture de l'incident.

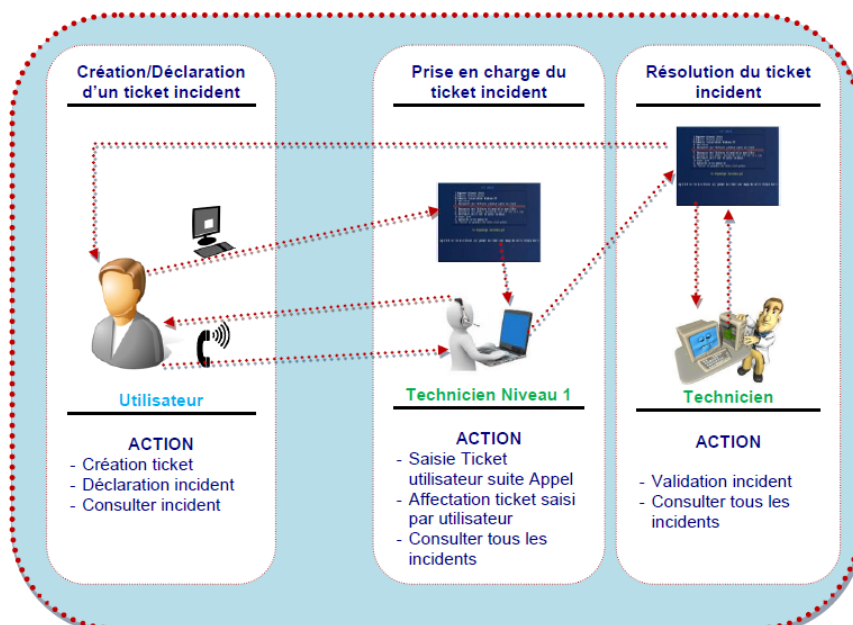
Les principales demandes sont :

- Dépannage,
- Installation logiciel,
- Problème messagerie,
- Problème impression,
- Problème lié au matériel (souris, clavier, écran...),
- Connexion réseau,
- Autres.

### 4. Workflow d'un ticket

Le Workflow (la représentation d'une suite d'opérations) d'un ticket peut se présenter ainsi:

- Déclaration du ticket par l'utilisateur via l'application ou déclaration du ticket par le *technicien 1* si la demande de l'utilisateur se fait par téléphone ;
- Attribution du ticket à un technicien ; Mail informant de la prise en charge ;
- Résolution du ticket par le *technicien 1* ou délégation à un technicien tiers ;
- L'utilisateur peut suivre l'état de son ticket via l'application.



## 5. Design

Le design devra permettre une bonne ergonomie. Le maximum d'informations sera affiché pour permettre une certaine clarté des informations. Des raccourcis devront être mis en place pour assurer une rapidité d'exécution.

## 6. Contraintes techniques

- Le langage utilisé pour cette solution est le C# (Visual C#).
- Une documentation technique des services mis en place dans la société GSB est disponible.

# III - Solution applicatives de gestion d'incidents

## 2. Utilisation de l'interface par l'utilisateur

Les utilisateurs peuvent appeler le service informatique pour faire leur demande. Le technicien niveau 1 chargé de prendre l'appel saisit l'incident en question et l'affecte à un technicien.

Il est prévu que les utilisateurs puissent renseigner eux-mêmes l'incident et l'enregistrer « si l'on se rend compte que la demande de travail n'est pas trop importante (ce qui évite d'avoir à immobiliser les techniciens derrière la ligne téléphonique) ». Le technicien niveau 1 n'aura plus à prendre les appels, il aura juste à affecter un technicien qui prendra en charge le ticket jusqu'à la clôture.

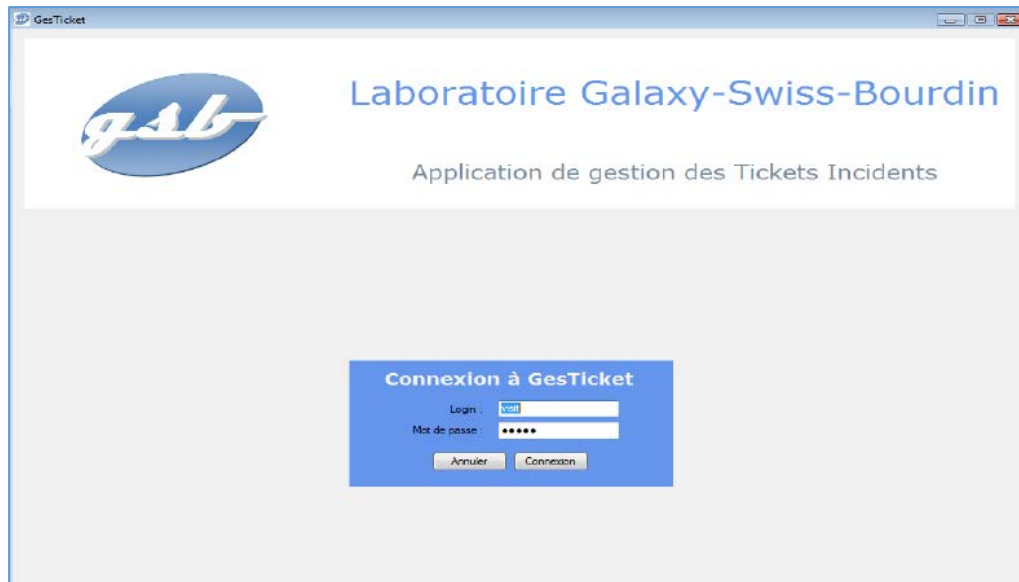
## 3. Identification

Un identifiant et un mot de passe est prévu pour chaque utilisateur. On distingue quatre types d'utilisateurs :

- Le gestionnaire,
- Les techniciens qui prennent en charge les incidents,
- Le technicien niveau 1 qui prend les demandes par téléphone, il les saisit y répond dans la mesure du possible sinon les affecte à un technicien
- Les utilisateurs (le personnel)

Les droits ne sont pas les mêmes pour tous les utilisateurs de l'application. Le nom, le prénom et le statut de l'utilisateur est rappelé sur toutes les vues.

La page d'identification est commune à tous les utilisateurs de l'application.



Après s'être identifié les utilisateurs arrivent sur la page d'accueil. Les deux boutons au-dessus du menu central leur permettent soit de créer un ticket incident soit de consulter leur(s) ticket(s) en cours.

En fonction du statut de l'utilisateur, les boutons du menu *change de couleur*. Pour les utilisateurs les boutons sont bleus alors que pour le personnel informatique (le gestionnaire, les techniciens et le technicien niveau 1), ils sont verts.

#### **4. Présentation de l'interface côté « utilisateur »**

##### ***a. Page d'accueil de l'utilisateur***



Laboratoire Galaxy-Swiss-Bourdin

Application de gestion des Tickets Incidents

Utilisateur : Jean Bonneau  
Statut : Visiteur

[Saisir incident](#)[Consulter incident](#)[Se déconnecter](#)



**b. Création d'un ticket par l'utilisateur**

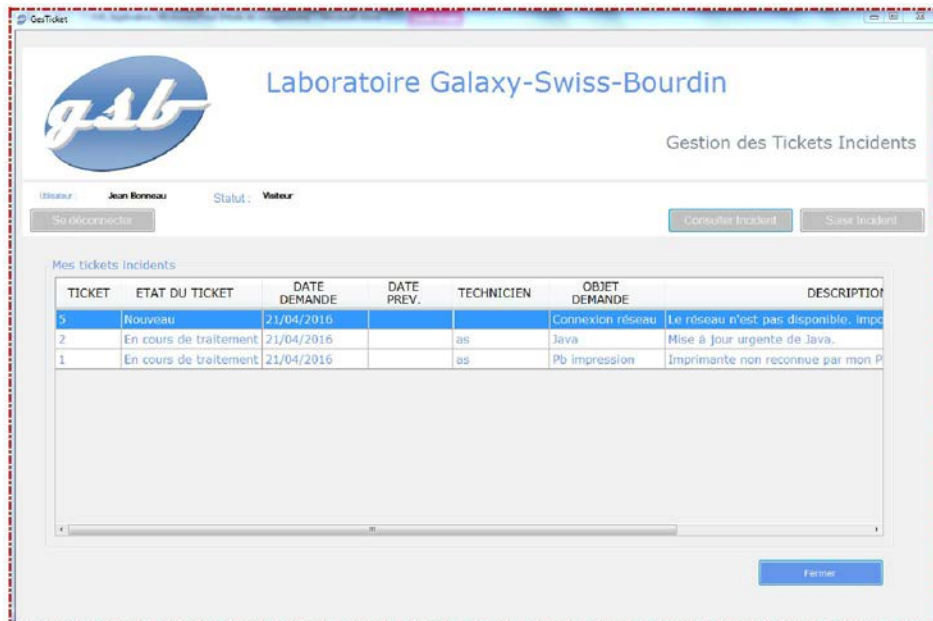
- L'utilisateur peut saisir et enregistrer son ticket dans « l'Application de gestion des tickets incidents ». Cette dernière a été conçue pour être simple et facile d'utilisation. Elle ne requiert pas une connaissance approfondie en informatique. Le nom de la personne connectée est rappelé sur chaque vue. (Cf la vue ci-dessous)

The screenshot shows a web browser window with the title 'GestTicket'. The page header features the 'gsb' logo and the text 'Laboratoire Galaxy-Swiss-Bourdin'. Below the header, the page title 'Gestion des Tickets Incidents' is displayed. A navigation bar shows the user 'Jean Bonneau' and their status 'Vendeur'. There are buttons for 'Se déconnecter', 'Consulter Incidents', and 'Saisir Incident'. The main section is titled 'Saisir nouvel incident'. It contains a dropdown menu for 'Utilisateur' with 'Bonneau Jean' selected, a date picker for 'Date de la demande' set to 'lundi 25 avril 2016', a text input for 'N° inventaire', a text area for 'Résumé', and a large text area for 'Description'. To the right of the 'Résumé' field is a button labeled 'Etat ticket : Nouveau'. At the bottom right are buttons for 'Enregistrer' and 'Annuler'.

- Il complète les différents champs qui sont obligatoire et enregistre son ticket, l'état du ticket est « Nouveau » par défaut. Dès que le technicien niveau 1 aura affecté le ticket à un technicien, il peut consulter l'avancé de son ticket via l'application.

### c. Consultation des tickets incidents

- L'utilisateur peut consulter son/ses ticket(s) en cours et ainsi suivre l'état de ses tickets.



## 5. Présentation de l'interface pour le « technicien »

### a. Page d'accueil



- Le Technicien niveau 1 a accès à 3 menus :
  - ✓ Créer incident
  - ✓ Affecter incident
  - ✓ Suivi des tickets



### b. Page « Créer incident »

- Un menu déroulant avec la liste des utilisateurs permet au technicien niveau 1 de choisir l'utilisateur pour lequel il crée un ticket incident.
- L'état pour chaque ticket est « Nouveau » par défaut.

### c. Page « Affecter incident »

- La liste des tickets dont l'état est « Nouveau » s'affiche.
- Le technicien niveau 1 choisit la ligne du ticket à affecter, par défaut le curseur est sur la 1ère ligne.
- Il clique sur le bouton « affecter » afin d'affecter un technicien (Cf les 2 vues suivantes).

TICKET	NOM - PRENOM UTILISATEUR	DATE	NUMERO INVENTAIRE	RESUME	DESCRIPTION
5	Jean Bonneau	21/04/2016	14 0007	Connexion réseau	Le réseau n'est pas disponible. Impossibilité

Il peut déterminer l'impact et la priorité de l'incident via des menus déroulants.

- Une liste déroulante lui permet de choisir le technicien auquel il affecte le ticket.
- Il doit déterminer une date de clôture prévisionnelle et une durée prévisionnelle.

The screenshot shows the 'Affecter ticket incident' dialog box. It has the following fields:

- Impact :** A dropdown menu.
- Priorité :** A dropdown menu.
- Choix du technicien :** A dropdown menu.
- Date de clôture prévue :** A date picker showing 'mardi 26 avril 2016'.
- Durée prévisionnelle :** A dropdown menu.

Buttons: 'Valider' (green) and 'Annuler' (green).

- L'état du ticket est modifié, dès lors il prend l'état « En cours de traitement ».

#### d. Page « Suivi des tickets »

The screenshot shows the 'Suivi des tickets' page. It features a table with the following data:

TICKET	ETAT TICKET	NOM - PRENOM UTILISATEUR	DATE DEMANDE	OBJET DEMANDE	TECHNICIEN	DATE PREV.	DATE CLOTURE INCIDENT
5	Nouveau	Bonneau Jean	21/04/2016	Connexion réseau	as		
8	En cours de traitement	Finck Jacques	25/04/2016	Test	as		
7	En cours de traitement	Finck Jacques	25/04/2016	Test	g2		
2	En cours de traitement	Bonneau Jean	21/04/2016	Java	as		
1	En cours de traitement	Bonneau Jean	21/04/2016	Pb impression	as		
4	Ticket clôturé	Tabone Sylvain	21/04/2016	Gestion des tickets	as		21/04/2016
3	Ticket clôturé	Chenu Laura	21/04/2016	Pb au démarrage	g2		21/04/2016

- L'ensemble des tickets s'affiche.

- Le menu déroulant permet de choisir par « état » les tickets à visualiser.
- En cliquant sur « afficher » la liste des tickets dont l'état a été choisi s'affiche.



## 6. Présentation de l'interface pour le « technicien »

### a. Page d'accueil



- Le Technicien a accès à 3 menus :
  - ✓ Affecter incident
  - ✓ Valider incident
  - ✓ Suivi des tickets

**b. Page « Affecter un incident »**

Voir formulaire « Affecter incident »

**c. Page « Valider incident »**

- La liste des tickets dont l'état est « En cours de traitement » s'affiche.
- Le technicien choisit la ligne du ticket à valider.
- Il clique sur le bouton « Valider incident » afin de valider un ticket.

TICKET	ETAT TICKET	NOM - PRENOM UTILISATEUR	DATE DEMANDE	DATE PREV.	OBJET DEMANDE	TECHNICIEN	DESCRIPTION INCIDENT
8	EN	Finck Jacques	25/04/2016		Test	as	Test 1
7	EN	Finck Jacques	25/04/2016		Test	g2	-----
2	EN	Bonneau Jean	21/04/2016		Java	as	Mise à jour urgente de Java.
1	EN	Bonneau Jean	21/04/2016		Pb impression	as	Imprimante non reconnue par mon

- Un menu déroulant lui permet de modifier l'état du ticket.
- Il choisit dans le calendrier la date de clôture.
- Il doit saisir la description de son intervention, expliquer la solution apportée.

**d. Page « Suivi tickets »**

Voir formulaire « Suivi des tickets »

**8- Présentation de l'interface pour le « gestionnaire »**

- Le gestionnaire a accès :
  - ✓ Saisir incident
  - ✓ Affecter incident
  - ✓ Valider incident
  - ✓ Suivi des tickets incidents





## Annexe 1 : Cahier Des Charges

### a. Définition du besoin

#### ➤ Définition de l'objet

Une cellule destinée à prendre en charge les demandes des visiteurs sera mise en place.

Par rotation, des personnels du service *Réseau & Système* seront affectés à une "ligne d'assistance" destinée à prendre par téléphone les doléances ou demandes de dépannage des visiteurs.

Cette cellule doit être équipée d'une solution lui permettant de prendre la demande avec le plus de précision en limitant au maximum les interrogations à l'interlocuteur.

#### ➤ Forme de l'objet

On souhaite un outil complet assurant :

- une gestion de parc dans laquelle seront consignées toutes les caractéristiques des équipements (processeur, mémoire, disque, logiciels installés lors de la délivrance du matériel), les éléments contractuels (date d'achat, garantie, fournisseur, etc.), leur affectation aux personnels visiteurs,
- une gestion des tickets d'incident, de leur enregistrement à leur résolution, avec une information au demandeur de l'avancement de la prise en compte,
- une possibilité de vision synthétique du fonctionnement de la cellule d'assistance : temps de traitement moyen des demandes, nombre d'appels, classification des demandes par niveau de complexité, volume d'intervention des différents techniciens, etc

### b. Contraintes

#### ➤ Utilisation

Le technicien doit opérer un minimum de saisie et pouvoir retrouver toutes les informations nécessaires à un diagnostic rapide sans chercher dans de trop nombreux écrans.

On ne souhaite pas affecter de personnel à la production intégrale de l'outil, mais des adaptations peuvent être faites pour répondre à des attentes spécifiques.

Les visiteurs doivent recevoir par mél un message à la prise en charge de la demande, et un autre lors de la résolution de l'incident. Cet échange doit être automatisable au maximum : l'envoi de message électronique doit se faire directement depuis l'interface de l'application sans passer par un autre outil.

L'outil ne sera pas accessible directement aux visiteurs dans un premier temps, mais il pourrait leur être ouvert par la suite si l'on se rend compte que la demande de travail n'est pas trop importante (ce qui évite d'avoir à immobiliser les techniciens derrière la ligne téléphonique). L'interface doit donc être simple d'utilisation, accessible et compréhensible à un non technicien, et capable de présenter l'évolution de la prise en charge de la demande.

## Annexe 2 : Fiche de prise en charge d'un incident

### Liste des informations à retenir

<b>Date/Heure</b>	Date et heure de la prise en compte de la demande.
<b>Demandeur</b>	Nom de la personne à l'origine de la demande.
<b>Equipement</b>	Nom unique du matériel concerné par la demande.
<b>Objet</b>	Nature de la demande
<b>Niveau</b>	<p><b>Degré de complexité de la demande</b></p> <ol style="list-style-type: none"> <li>1. Demande basique (niveau par défaut), ne nécessite pas une intervention logicielle ou matérielle, dépannage en direct avec l'interlocuteur. <i>Exemple : perte d'identifiant de messagerie, déblocage d'un rapport verrouillé, etc.</i></li> <li>2. Demande nécessitant une intervention à distance.</li> <li>3. Demande nécessitant une intervention matérielle ou une réinstallation logicielle lourde.</li> <li>4. Demande nécessitant l'adaptation du site Web de gestion des Comptes rendus et des frais de visite.</li> </ol>
<b>Technicien</b>	Nom du technicien prenant en charge la demande. Pour le niveau 1, il s'agit de la personne qui répond au téléphone.
<b>Date intervention</b>	Date de la prise en charge effectif du problème. Pour le niveau 1, c'est la date de l'appel.
<b>Solution</b>	Rédaction de la solution apportée (changement de matériel, mise à jour d'une version, etc).
<b>Durée</b>	Temps effectif en minutes de l'intervention. Pour le niveau 1, on retiendra la durée d'appel.

## Annexe 3 : Elaboration de classes métier

*Un début d'élaboration de classes métier a été mené. Vous êtes libre de vous en inspirer, de le modifier.*

```
public class Visiteur
{
    private int idVisiteur;
    private String nomVisiteur;
    private String prenomVisiteur;
    .....;
    //Prévoir un constructeur, des accesseurs
}

public static class Passerelle
{
    private static SqlConnection laConnection;

    // Permet la connexion à la base de données
    public static void seConnecter()

    // Recherche dans la table Matériel le Produit l'identifiant
    // unId. Renvoie l'objet Produit correspondant, null si non
    // trouvé
    public static Produit getProduit(int unId)

    // Recherche dans la table Visiteur le Visiteur d'identifiant
    // unId. Renvoie l'objet Visiteur correspondant, null si non
    // trouvé
    public static Visiteur getVisiteur(int unId)

    // Recherche dans la table Visiteur le Visiteur de log unLog,
    // de mot de passe unMotPasse. Renvoie l'objet Visiteur
    // correspondant, null si non trouvé
    public static Visiteur getVisiteur
        (String unLog, String unMotPasse)

    //Recherche dans la table Ticket le ticket d'identifiant
    // unId. Renvoie l'objet Ticket correspondant, null si non
    // trouvé
    public static Ticket getTicket (int unId)

    // Recherche dans la table Ticket les objets Produit dont
    // l'objet de référence est l'objet Produit passé en
    // paramètre.
    public static List<Ticket> getlesTicketss
        (Produit leProduitDeReference)

    //Renvoie l'ensemble des objets Tickets dont la date de
    //début est supérieur à la date du jour, concernant un objet
    //Produit passé en paramètre
    public static List<Ticket> getTicketAVenir
        (Produit unTicket)
}
```