# Projet Base de Données : Bibliothèque

Etapes de création, chargement de la base	2
Fonctionnalités	2
Hypothèses	2
Fonctionnalités supplémentaires	3
Scénarios de test pour chaque fonction	3
Panel de Connexion du logiciel	3
Menu panel Admin	4
Panel Gestion Livre	4
Panel gestion etudiant	g
Panel Retard	10
Panel Etudiant	10

## Etapes de création, chargement de la base

Nous avons réfléchi au script de création fourni en PJ et aux différents types d'attributs et clés à déclarer, puis nous l'avons exécuté sur la base de données de l'iut. Nous avons ensuite utilisé les données du site data.gouv.fr pour donner de la matière à nos affichages de livre en important le .csv sur sql developer. Puis nous avons fait le lien entre Eclipse, notre IDE de java et le langage sql avec JDBC et notamment le driver en .jar. Nous nous connectons à la BD via un tunnel ssh. Une fois qu'on pouvait se connecter, on a créé le squelette du projet et appris à utiliser l'extension Egit pour le travail collaboratif, et on était partis.

#### **Fonctionnalités**

Les étudiants peuvent:

- Rechercher des livres selon différents critères
- Réserver au maximum 5 livres

Les bibliothécaires peuvent:

- Rechercher des livres selon différents critères
- Prêter un livre à un étudiant avec une limite de 5 en même temps
- Récupérer un livre rendu
- Gérer les comptes étudiants
- Gérer les livres
- Afficher les emprunts en retard

## Hypothèses

- Un élève ne peut pas émettre plus de 5 réservations en même temps

```
create trigger CHECK_RESERV_LIMIT

    before insert
    on RESERV
    for each row

DECLARE
    nb_reserv number;

BEGIN
    SELECT COUNT(*) INTO nb_reserv FROM RESERV WHERE ID_ET = :new.ID_ET AND DATE_FIN_RES IS NULL;
    if nb_reserv >= 5 THEN -- erreur si le nombre de reservation était à 5 car il va passer à 6 sinon
        raise_application_error(-20000, 'Nombre de réservation maximum atteint');
    end if;

END;
/
```

- Un élève ne peut pas emprunter plus de 5 exemplaires en même temps

```
create trigger CHECK_EMPRUNT_LIMIT
    before insert
    on EMPRUNT
    for each row

DECLARE
    nb_emprunt number;

BEGIN
    SELECT COUNT(*) INTO nb_emprunt FROM EMPRUNT WHERE ID_ET = :new.ID_ET AND DATE_RETOUR IS NULL;
    if nb_emprunt >= 5 THEN -- erreur si le nombre d'emprunt était à 5 car il va passer à 6 sinon
        raise_application_error(-20000,'Nombre d emprunt maximum atteint');
    end if;

END;
/
```

 Les exemplaires en retard restent dans la BD, c'est à la bibliothécaire de fixer la date de fin d'emprunt (date d'emprunt reste nulle si le livre est en cours d'emprunt)

- Il est possible de réserver un livre dont tous les exemplaires sont empruntés, pour que l'étudiant puisse emprunter le livre dès que ses camarades en auront rendu un exemplaire
- Il est possible d'emprunter sans réserver à l'avance
- Plusieurs contraintes d'intégrité: La suppression d'un livre entraîne la suppression de toutes les réservations, exemplaires et donc emprunts qui lui sont liés; La suppression d'un étudiant entraîne la suppression de ses emprunts et réservations. Exemple ici de la déclaration d'une contrainte d'intégrité pour qu'un emprunt se supprime lorsqu'un exemplaire est supprimé

```
alter table emprunt drop constraint SYS_C0046893;
alter table emprunt add
   constraint FK_EMPRUNT_EXEMPLAIRE
   foreign key (ID_EX)
   REFERENCES EXEMPLAIRE(ID_EX)
   ON DELETE CASCADE;
```

# Fonctionnalités supplémentaires

La bibliothécaire peut trier par nom de colonne et par ordre croissant ou décroissant les étudiants

La bibliothécaire peut choisir le nombre d'exemplaire et le modifier pour chaque livre La bibliothécaire peut voir les réservations et les emprunts du livre sélectionné Affiche la disponibilité, le nombre d'exemplaire d'un livre On peut pas prêter un livre qui n'a pas d'exemplaire disponible On ne peut pas rendre un livre qui n'est emprunté par aucun étudiant

# Scénarios de test pour chaque fonction

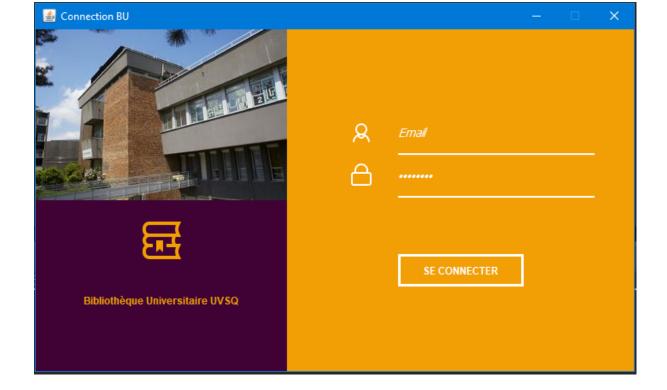
#### Panel de Connexion du logiciel

Le système vérifie si l'email et le mot de passe correspondent à ceux de l'admin sinon contact la base de données pour trouver un étudiant avec les mêmes coordonnées.

Si les données saisie ne correspondent pas envoie un message d'erreur

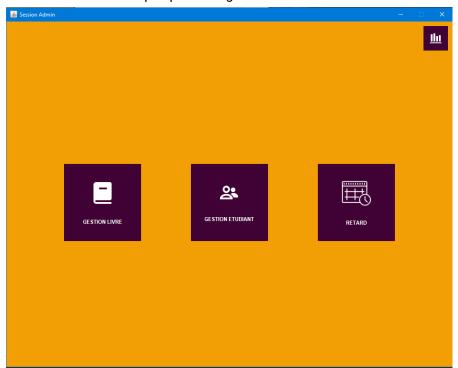
email admin : a mdp admin : a

email etudiant : damien mdp etudiant :azerty



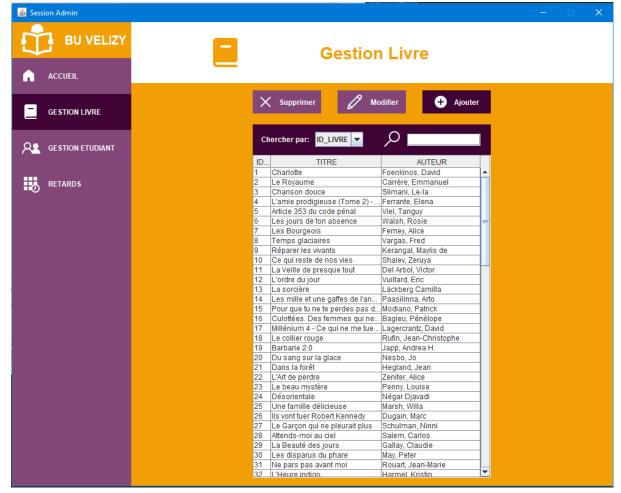
## Menu panel Admin

Permet de choisir sur quel panel de gestion aller



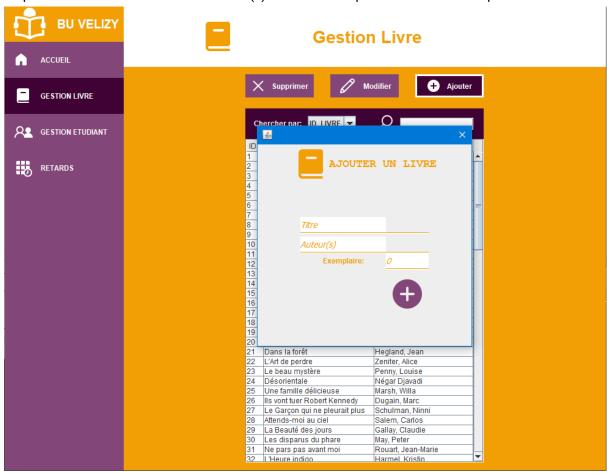
#### Panel Gestion Livre

Le panel tel quel permet uniquement de chercher un livre par les différentes colonnes de la table LIVRE ainsi que d'ajouter un livre

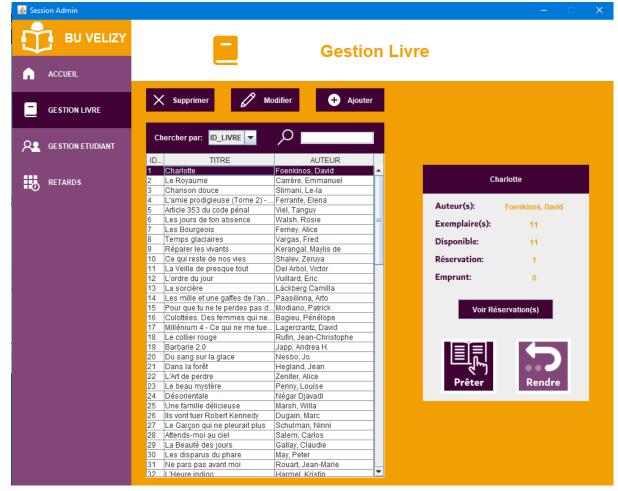


Si on clique sur Ajouter

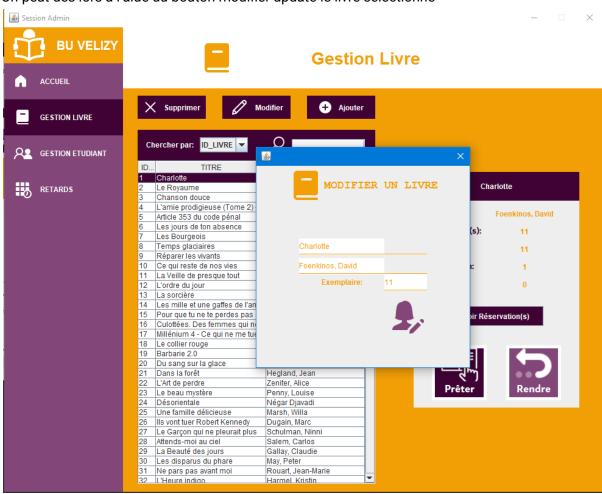
On peut dès lors choisir le titre et auteur(s) du livre ainsi que le nombre d'exemplaire de celui ci



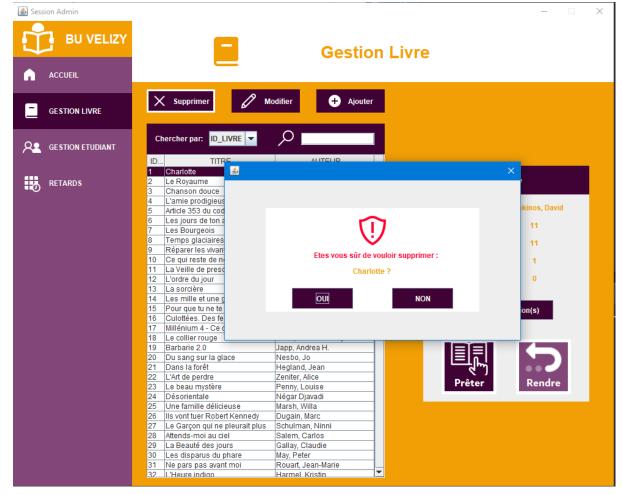
Une fois un livre sélectionné dans le tableau en cliquant sur la ligne correspondante le panel change:



On peut dès lors à l'aide du bouton modifier update le livre sélectionné



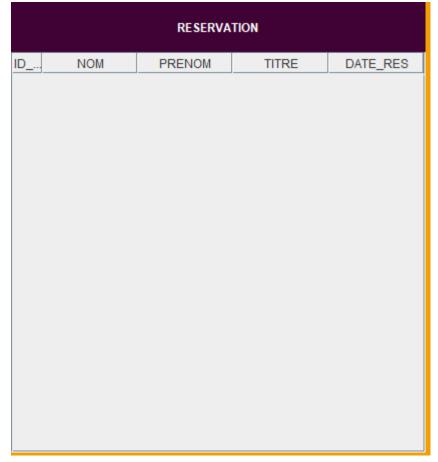
Ou encore supprimer le livre:



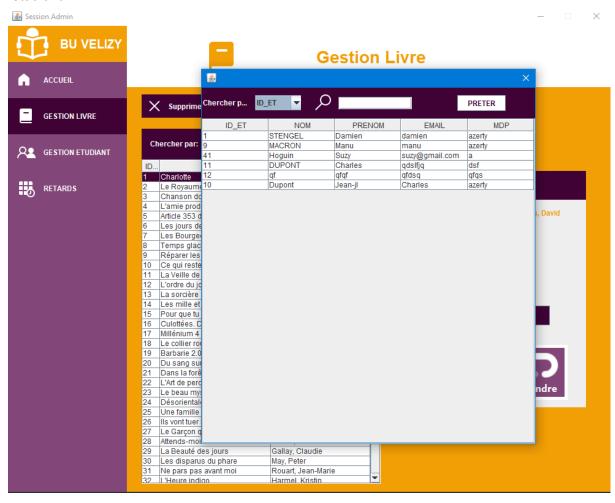
De plus le panel à droite indique toute les informations nécessaire à la bibliothécaire



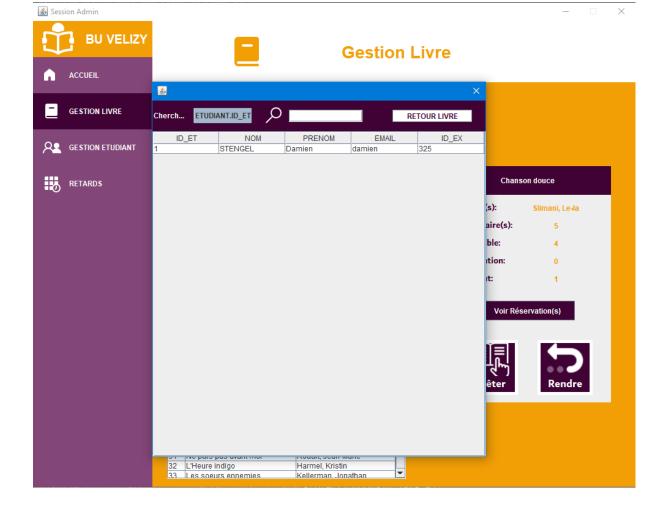
L'admin peut alors voir les réservations en cours sur le livre:



Ainsi que prêter un livre si il a un exemplaire de disponible avec la possibilité de rechercher l'étudiant:

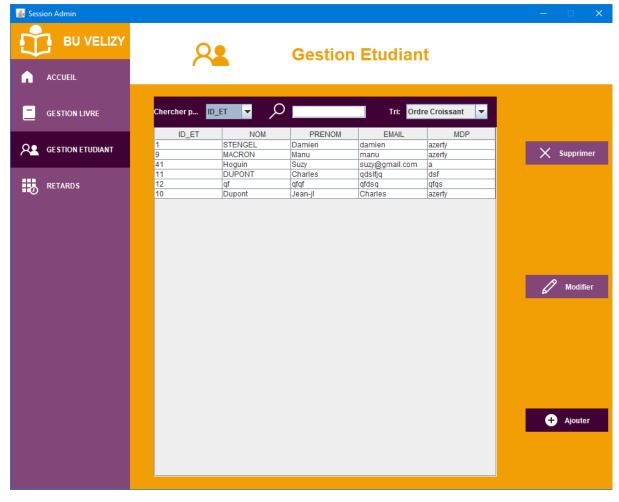


Enfin l'admin peut rendre un livre si il a été emprunté et non rendu:



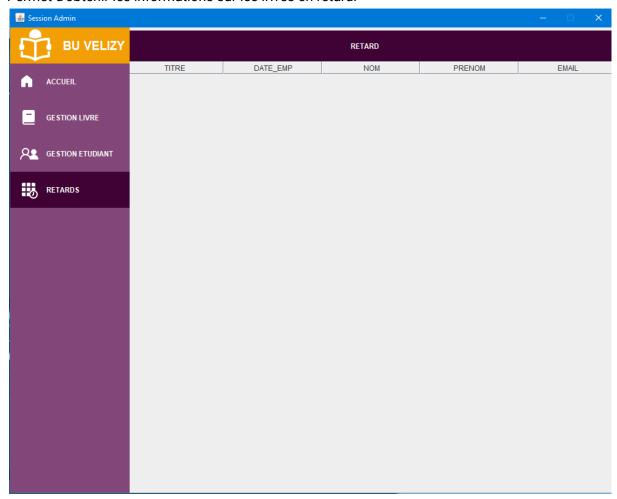
## Panel gestion etudiant

Gestion Etudiant a les mêmes fonctionnalités de suppression, modification et d'ajout. Il permet aussi une recherche par ordre croissant ou décroissant des colonnes.



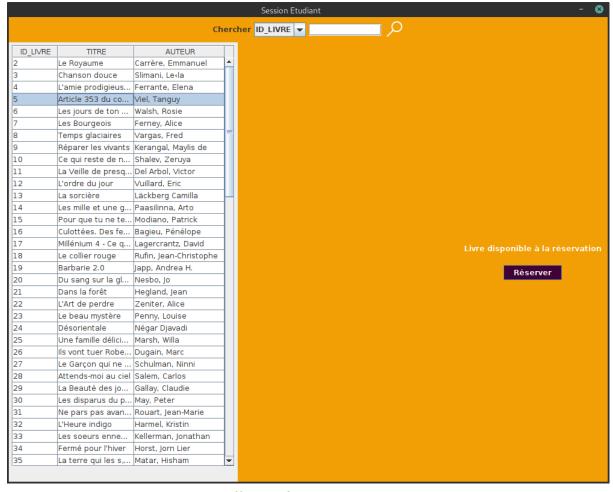
### Panel Retard

Permet d'obtenir les informations sur les livres en retard.

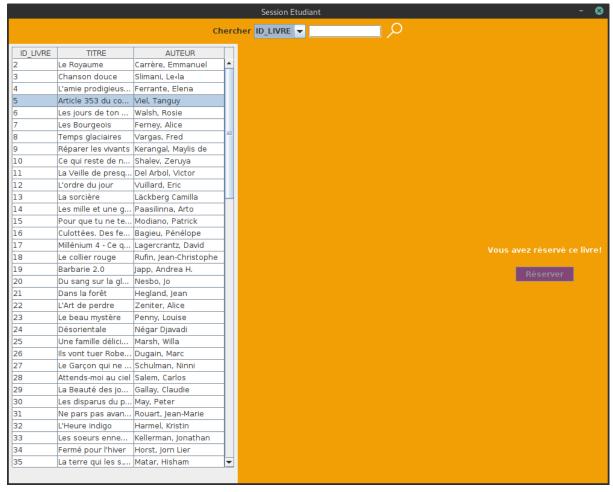


#### Panel Etudiant

Lorsque un étudiant met ses identifiants dans le panel de connexion, il est redirigé vers son propre panel:



Il peut y rechercher des livres avec différents filtres et les sélectionner. Si je clique sur réserver ici par exemple:



Et dans le cas où le livre est déja réservé par quelqu'un d'autre:

