

htw saar

RAPPORT DE PROJET GESTION D'UNE RESIDENCE UNIVERSITAIRE

PAR

- Tom Naudin (Gestion, FR)
- Christopher Yepmo (Informatique, CMR)
- Eva-Johanna Herrlich (Gestion, GER)

Professeur

Sorin Stratulat



TABLE DES MATIERES

Le sujet	3
Dictionnaire des données	4
MCD	5
Modèle relationnel	5
Entités	5
Association	6
Liste des tables sur ACCESS	7
Exemples de formulaire	11
Requêtes	14
Conclusion	14
Bibliographie	17

Le sujet

Nous avons choisi ce thème car nous sommes tous des étudiants et la plupart d'entre nous habitons dans des résidences estudiantines et nous avons beaucoup de connaissances par rapport à ce type de logements.

Les résidences universitaires que notre application permettra de gérer sont les résidences CROUS situées à Metz (comme celle du Saulcy, de Bridoux ou du Technopôle). Chaque résidence est située à une station qui a un nom, une surface et un nombre de bâtiments spécifique. Dans chacune de ces stations travaillent des employés, qui sont identifiés par leurs noms et prénoms, leur date de naissance, leurs horaires de travail, leur sexe, leur contact et leur poste. Les employés peuvent travailler dans plusieurs bâtiments et/ou établissements. Les établissements ont un nom, des horaires d'ouvertures et des informations de contact, tandis que les bâtiments n'ont qu'un nom et/ou un numéro.

Dans chaque bâtiment, on retrouve plusieurs chambres occupées par des étudiants et l'on cherchera à connaître le numéro de cette chambre, sa superficie, le type de chambre que c'est (si c'est une colocation ou autre chose) et le loyer payé par l'étudiant. Le système informatique aimerait également savoir si la chambre est occupée ou si celle-ci n'est que réservée. Dans le cas d'une occupation, les étudiants doivent, pour des raisons d'assurance donner leur nom, prénom, les dates d'arrivée et de départ, leur contact, leur niveau scolaire et leur IBAN. Ils doivent également fournir leur assurance habitation comprenant leur nom, la date de validité de cette assurance, leur contact et leur adresse. Enfin, pour attester qu'ils sont étudiants, la résidence leur demandera l'université dans laquelle ils étudient avec le nom de l'école, l'adresse et le nom du directeur de l'université.

Si la chambre n'est que réservée, l'administration souhaiterait savoir le numéro et la date de la réservation, le nombre de chambre réservées et le temps que cela durera.



Un étudiant obtient également des aides aux logement et il faudrait que la résidence connaisse le type d'aide, l'adresse et le contact de l'organisation qui apporte l'aide et le montant de cette dernière.

Enfin, les différentes stations CROUS possèdent un système de communication qui permet aux employés et aux étudiants d'envoyer des messages et la date de leur envoi pour stipuler des problèmes rencontrés (par exemple une panne). Les employés pourront donc appartenir à une équipe de réparation qui se chargera de la panne et d'expliquer le type de panne et son coût de réparation.

Dictionnaire des données

Horaire_travail: Heures de travail

Date nais: Date de naissance

Nombre Bâtiment: Nombre de bâtiment dans une station

Contact: Informations contact

Numéro: Numéro de chambre

Superficie: Superficie de chambre

TypeCham: Type de chambre

DateArr: Date d'arrivée

DateDep: Date de départ

Type_aide: Type de l'aide (CAF)

Niv_etude: Niveau scolaire

Assurance_habitation: Assurance d'habitation d'un etudiant

NumRes: Numéro de réservation

DateRes: Date de réservation

TempsRestant: Temps qui reste pour valider la réservation

Sujet: Sujet de message

Message: Message

Date_env: Date d'envoie

Type_panne: Type de panne

Coût_repar: Coût de réparation



Equipe_reparation: Equipe de réparation

Date_payement: Date de paiement

Nom: Nom d'étudiant

Modèle Conceptuel de Donnée

Il se trouve en annexe de ce document (mcd residence universitaire.pdf)

Modèle relationnel des Données Entités

Horaire_travail (Id horraire, Lundi, Mardi, Mercredi, Jeudi, Vendredi, Samedi, Dimanche)

Bâtiment (Id bat, Nom, Nombre_etage, Nombre_chambre, #Id_station)

Panne (Id panne, Type panne, Coût repar)

Equipe_reparation (Id_eqp, Entreprise, Nbre_personnel)

Station (Id station, Nom, Superficie, Nombre bâtiment, Adresse)

Employé (Matricule, Nom, Prénom, Date_nais, Sexe, Contact, Fonction)

Chambre (<u>Numéro</u>, HO, Superficie, TypeCham, Loyer, #Id_bat, #N°_INE, NumRes, DateRes, TempsRestant, Caution)

Etablissement (<u>Id_etabl</u>, Rôle, HD, Info_contact, #Id_station, #id_horaire)

Etudiant (N° INE, Nom, Prenom, DateArr, DateDep, Contact, Email, Niv_etude, IBAN, N°_chambre)

Assurance_habitation (Id_assur, Nom_compagnie, Contact, Adresse)

Aide_logement (<u>Id_aide</u>, Type_aide, Contact, Montant, Adresse)



Adresse (Id_add, Numéro, Nom_voie, Code_postale, Ville, Pays)

Communication (Id-com, Sujet, Message, Date_env)

Université (Nom, Directeur, #Id_add)

Association

Dans (_Id_bat, Id_panne)

Reparer (Id eqp, Matricule, Id_panne)

Travailler_etabl (Matricule, Id_etabl, Est_directeur)

Vivre à (Matricule, Id add)

Recevoir_Empl (Matricule, Id_com)

Envoyer_Empl (Matricule, Id_com)

Recevoir Etud (N° INE, Id com)

A (<u>Id aide</u>, <u>Id assur</u>, N°_INE)

Etudier (<u>N° INE</u>, Nom)

Payer (_N°_INE, _Id_station, montant, Motif, Date_payement)



Exemples de table sur ACCESS

Pour mener à bien le projet, nous avons dû créer une trentaine de table que nous avons remplis avec des données fictives pour effectuer nos tests. Cidessous se trouve des exemples.

Table Chambre:

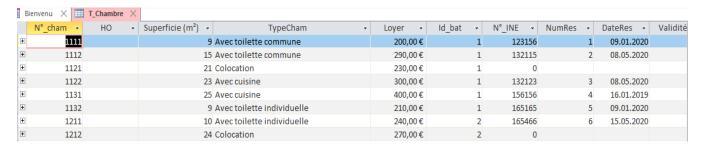


Table Etablissement:

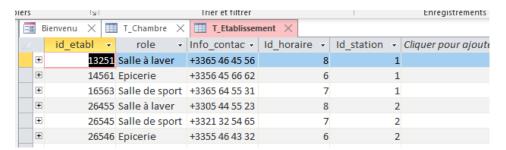


Table Étudiant :

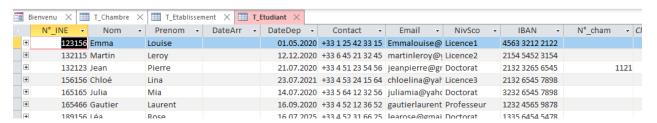


Table Horaire_travail:



Table Panne:



Table Station:

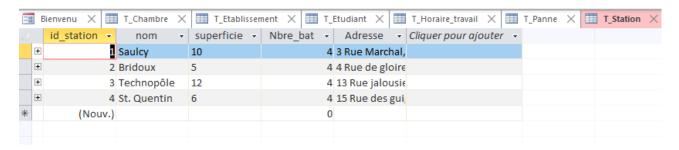


Table Université:



Table Batiment:

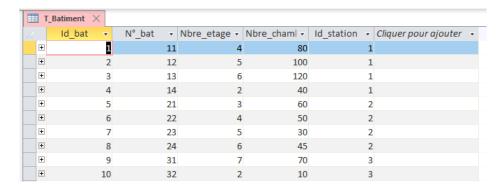


Table Équipe de réparation :

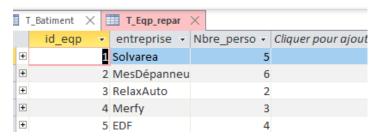


Table Employé:



Table assurance habitation:

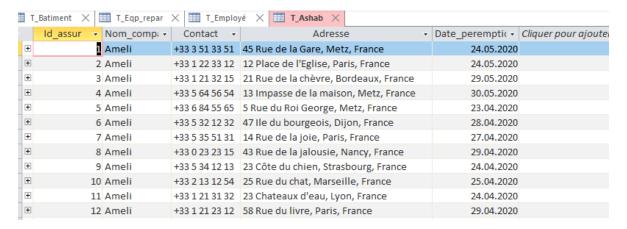


Table Aide Logement:

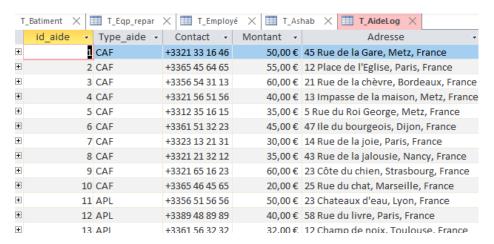
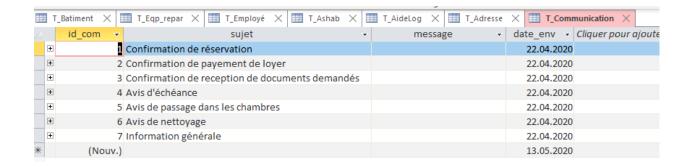


Table Adresse:



Table Communication:



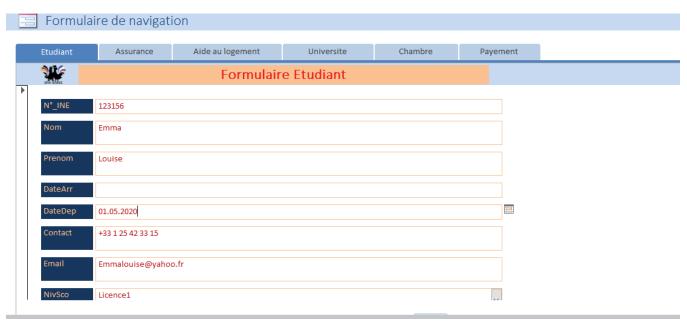
Formulaires

Pour les formulaires nous avons créé trois types de formulaires. Un formulaire de démarrage qui se présente à l'ouverture de l'application et contenant des boutons vers les différents formulaires de navigation. Nous avons 4 formulaires de navigation (Employés, étudiants, pannes, stations). Ces formulaires de navigation contiennent des formulaires pour nous permettre la modification des données de toutes les tables. Ci-dessous se trouvent quelques exemples des différents formulaires.

Formulaire de Démarrage contenant des boutons vers les formulaires de navigation :



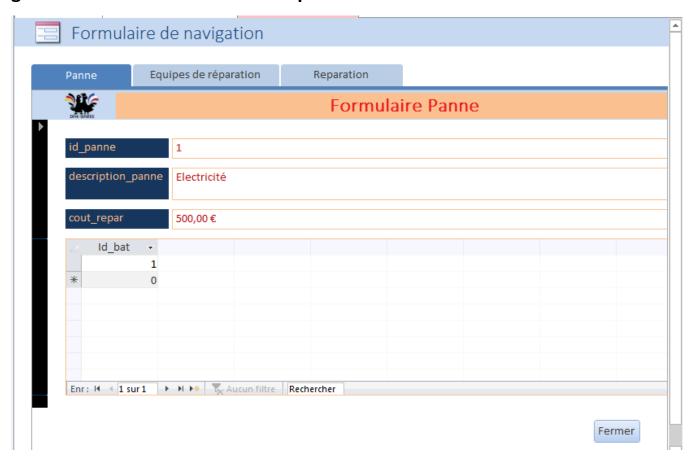
Formulaire de navigation « étudiant » contenant les autres formulaires pour la gestion des informations sur les étudiants :



Formulaire de navigation « employé » contenant les formulaires pour la gestion des informations des employés :

٥	-=	Formula	aire de navigatio	n				
		Employé	Adresse	Messages	Envoyer message	Horaire de travail	Etablissement	
		Die States						
	•	Matricule	1					
		Nom	Jean					
		Prénom	Pierre					
		Date_nais	30.03.1990					
		Sexe	М			~		
		Contact	+33 1 25 42 33 15					
		Fonction	Chef			~		

Formulaire de navigation « Panne » contenant les formulaires pour la gestion des informations sur les pannes :



Formulaire de navigation « Station » contenant les formulaires pour la gestion des informations sur les stations :

	Formula	aire de navigatio	n		_
	Station	Batiment	Payement		
	DEN STATES			T_Station	
•	id_station	1			
	nom	Saulcy			
	superficie	10			
	Nbre_bat	4			
	Adresse	3 Rue Marchal, Metz, Fr	ance		



Requêtes

Comme listé dans notre cahier de charge lors de la première présentation en début de projet, ci-dessous se trouve les requêtes que nous devrions effectuer.

1. Quels sont les employés qui travaillent à une certaine station ?

 ${\tt SELECT\ T_Station.id_station,\ T_Employ\'e.Matricule,\ T_Employ\'e.Nom,\ T_Employ\'e.Pr\'enom}$

FROM (T_Station INNER JOIN T_Etablissement ON T_Station.[id_station] = T_Etablissement.[Id_station]) INNER JOIN (T_Employé INNER JOIN R_Travailler_etabl ON T_Employé.[Matricule] = R_Travailler_etabl.[Matricule]) ON T_Etablissement.[id_etabl] = R_Travailler_etabl.[Id_etabl];

2. Quelles sont les assurances habitation qui ne sont plus valide et à qui est-ce qu'elles appartiennent ?

SELECT R_A.Id_assur, R_A.N°_INE, T_Ashab.Date_peremption, T_Etudiant.Nom, T_Etudiant.Prenom

FROM T_Etudiant INNER JOIN (T_Ashab INNER JOIN R_A ON T_Ashab.Id_assur = R_A.Id_assur) ON T_Etudiant.N°_INE = R_A.N°_INE

WHERE (((T_Ashab.Date_peremption) < Date()));

3. Quels bâtiments, sur quels sites ont quelles pannes?

SELECT R_Dans.Id_bat, R_Dans.Id_panne, T_Batiment.Id_bat, T_Station.id_station, T_Station.nom

FROM T_Panne INNER JOIN ((T_Batiment INNER JOIN T_Station ON T_Batiment.Id_station = T_Station.id_station) INNER JOIN R_Dans ON T_Batiment.Id_bat = R_Dans.Id_bat) ON T_Panne.id_panne = R_Dans.Id_panne;

4. Quels employés font partir d'une certaine équipe de réparation ?

SELECT R_Reparer.Id_eqp, R_Reparer.Matricule, T_Employé.Nom, T_Employé.Prénom

FROM T_Eqp_repar INNER JOIN (T_Employé INNER JOIN R_Reparer ON T_Employé.Matricule = R_Reparer.Matricule) ON T_Eqp_repar.id_eqp = R Reparer.Id eqp

WHERE (((R Reparer.Id eqp)=1));



5. Quelles chambres sont encore disponibles sur le site XY?

SELECT T_Etudiant.N°_cham, T_Chambre.N°_cham, T_Chambre.NumRes, T_Station.id_station

FROM ((T_Batiment INNER JOIN T_Station ON T_Batiment.Id_station = T_Station.id_station) INNER JOIN T_Chambre ON T_Batiment.Id_bat = T_Chambre.Id_bat) LEFT JOIN T_Etudiant ON T_Chambre.N°_cham = T_Etudiant.N°_cham

WHERE (((T_Etudiant.N°_cham) Is Null) AND ((T_Chambre.NumRes) Is Null));

6. Combien d'étudiants déménageront en Juillet?

SELECT T_Etudiant.DateDep, T_Etudiant.N $^\circ$ _INE FROM T_Chambre LEFT JOIN T_Etudiant ON T_Chambre.N $^\circ$ _cham = T_Etudiant.N $^\circ$ _cham WHERE (((T_Etudiant.DateDep) Like "*/07/*"));

7. Quel étudiant n'a pas encore payé le loyer pour le mois XY?

SELECT R_Payer.Motif, T_Etudiant.N°_INE, T_Etudiant.Nom, T_Etudiant.Prenom, T_Etudiant.N°_cham

FROM T_Etudiant LEFT JOIN R_Payer ON T_Etudiant.N°_INE = R_Payer.N°_INE WHERE (((R_Payer.Motif)<>"Loyer")) OR (((R_Payer.Motif) Is Null));

8. Quel est le numéro de réservation d'un certain étudiant?

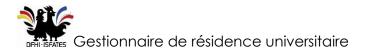
SELECT T_Chambre.NumRes, T_Chambre.N°_INE

FROM T_Chambre LEFT JOIN T_Etudiant ON T_Chambre.N°_cham = T_Etudiant.N°_cham WHERE (((T_Chambre.NumRes) Is Not Null));

9. Combien d'étudiants de l'université A vivent dans la résidence X?

SELECT T_Etudiant.N°_INE, T_Etudiant.Nom, T_Etudiant.Prenom, R_Etudier.Nom_uni, T_Chambre.Id_bat, T_Chambre.N°_cham, T_Station.nom

FROM T_Universite INNER JOIN ((((T_Station RIGHT JOIN T_Batiment ON T_Station.id_station = T_Batiment.Id_station) RIGHT JOIN T_Chambre ON T_Batiment.Id_bat = T_Chambre.Id_bat) RIGHT JOIN T_Etudiant ON T_Chambre.N°_cham = T_Etudiant.N°_cham) INNER JOIN R_Etudier ON



T_Etudiant.N°_INE = R_Etudier.N°_INE) ON T_Universite.Nom_uni = R Etudier.Nom uni;

10. Combien de chambres ont été réservées pour le mois de janvier?

SELECT T_Chambre.N°_cham, T_Etudiant.N°_INE, T_Chambre.NumRes, T_Chambre.DateRes

FROM T_Etudiant INNER JOIN T_Chambre ON T_Etudiant.N°_INE = T_Chambre.N°_INE WHERE (((T_Chambre.DateRes) Like '*/01/*'));

11. Qui est le directeur de l'université dans laquelle étudie l'étudiant Y?

SELECT T_Universite.Directeur, R_Etudier.Nom_uni, R_Etudier.N°_INE, T_Etudiant.Nom, T_Etudiant.Prenom

FROM T_Universite INNER JOIN (T_Etudiant INNER JOIN R_Etudier ON T_Etudiant.N°_INE = R_Etudier.N°_INE) ON T_Universite.Nom_uni = R_Etudier.Nom_uni

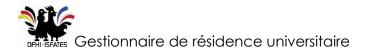
WHERE (((T_Etudiant.N°_INE)=165466));

12. Combien a coûté la réparation de la panne dans l'établissement X?

SELECT T_Batiment.Id_bat, R_Reparer.Id_panne, T_Panne.cout_repar

FROM (T_Panne INNER JOIN R_Reparer ON T_Panne.id_panne = R_Reparer.Id_panne) INNER JOIN (T_Batiment INNER JOIN R_Dans ON T_Batiment.Id_bat = R_Dans.Id_bat) ON T_Panne.id_panne = R_Dans.Id_panne

WHERE (((T_Batiment.Id_bat)=1) AND ((R_Reparer.Id_panne)=1));



Conclusion

Nous sommes trois étudiants de nationalités différentes. À cause de la situation que nous vivons actuellement (COVID-19) il était plus difficile de rester en contact et chacun avait également d'autres travaux à réaliser.

Toutefois la communication a bien fonctionné. Nous avons apprécié travailler sur ce projet car le thème était libre. Cela nous a permis d'associer le travail au sujet qui nous est très familier, une résidence universitaire. Comme beaucoup d'étudiants vivent, au moins pour le début de leurs études dans une résidence universitaire, il n'était pas difficile de décider.

Nous avons rencontré des difficultés sur Access, comme c'était la première fois de l'utiliser. Quand nous étions encore en présentiel à Metz nous avions déjà élaboré le MCD et cela aurait été difficile à distance puisqu'il faut que le MCD et les relations sur Access soient cohérent et que chacun comprenne le processus.

Qui a fait quoi

Eva-Johanna: Mise en page de la présentation, formulation du sujet, contribué au MCD, certaines requêtes, contribué au dictionnaire des données

Tom: Formulation du sujet, Contribution au MCD, participation au modèle relationnel de données, certaines tables Access

Christopher Yepmo: Formulation du projet, création et dessin du MCD dans l'ordinateur (avec lucidchart), conversion en MRD, Création des tables et remplissage avec des données pour test, création des relations entre les tables, création des requêtes, création des formulaires, rédaction et mise en page du rapport.

Bibliographie

https://support.office.com/en-us/access

https://www.youtube.com/playlist?list=PLSdtjcTY3VgRyfCNwJBvcLMGCtpUXvt9I

