

BTS SERVICES INFORMATIQUES AUX ORGANISATIONS	SESSION 2022
ANNEXE 7-1-B : Fiche descriptive de réalisation professionnelle (recto)	
Épreuve E5 - Conception et développement d'applications (option SLAM) - Coefficient 4	

DESCRIPTION D'UNE RÉALISATION PROFESSIONNELLE		N° réalisation :				
Nom, prénom : SAULNIER GUILLAUME		N° candidat : 02144598174				
Épreuve ponctuelle <input checked="" type="checkbox"/>	Contrôle en cours de formation <input type="checkbox"/>	Date : 17 / 02 / 2022				
Contexte de la réalisation professionnelle La MAISON DES LIGUES DE LA LORRAINE (M2L), établissement du Conseil Régional de Lorraine, est responsable de la gestion du service des sports et en particulier des ligues sportives ainsi que d'autres structures hébergées. La M2L doit fournir les infrastructures matérielles, logistiques et des services à l'ensemble des ligues sportives installées. M2L souhaiterait mettre en place un service en ligne de gestion de réservations de salles de réunion.						
Intitulé de la réalisation professionnelle Développement d'une plate-forme en ligne de réservation de salle.						
Période de réalisation : 2021/2022 Lieu : Groupe GEFOR Paris 1 ^e Modalité : <input type="checkbox"/> Seul(e) <input checked="" type="checkbox"/> En équipe						
Compétences travaillées <input checked="" type="checkbox"/> Concevoir et développer une solution applicative <input checked="" type="checkbox"/> Assurer la maintenance corrective ou évolutive d'une solution applicative <input checked="" type="checkbox"/> Gérer les données						
Conditions de réalisation¹ (ressources fournies, résultats attendus) <table border="0"> <tr> <td>Ressources fournies :</td> <td>Résultats attendus :</td> </tr> <tr> <td> <ul style="list-style-type: none"> Cahier des charges M2L </td> <td> <ul style="list-style-type: none"> Application Web de réservation de salle <ul style="list-style-type: none"> Inscription et gestion de compte utilisateur Réserver une salle Ajout de participants Création de tickets de réclamation Documentation de mise en service </td> </tr> </table>			Ressources fournies :	Résultats attendus :	<ul style="list-style-type: none"> Cahier des charges M2L 	<ul style="list-style-type: none"> Application Web de réservation de salle <ul style="list-style-type: none"> Inscription et gestion de compte utilisateur Réserver une salle Ajout de participants Création de tickets de réclamation Documentation de mise en service
Ressources fournies :	Résultats attendus :					
<ul style="list-style-type: none"> Cahier des charges M2L 	<ul style="list-style-type: none"> Application Web de réservation de salle <ul style="list-style-type: none"> Inscription et gestion de compte utilisateur Réserver une salle Ajout de participants Création de tickets de réclamation Documentation de mise en service 					
Description des ressources documentaires, matérielles et logicielles utilisées² <u>Base de données :</u> SGBD MariaDB v15, logiciel DBeaver et Postman <u>Développement côté serveur :</u> environnement Node.js v16.13.1, gestionnaire de paquet NPM v8.3.0 <u>Développement côté client :</u> cadre applicatif React.js v17.0.2 <u>Outil de développement :</u> IDE VSCode <u>Outils collaboratifs :</u> outils de versionnage Git et GitHub, Google Sheets, Figma						
Modalités d'accès aux productions³ et à leur documentation⁴ <u>Production :</u> Lien Github -> https://github.com/BSevault/PPE_M2L_backend et https://github.com/BSevault/PPE_M2L_frontend <u>Documentation :</u> La documentation se trouve respectivement sur chaque repo GitHub. L'ensemble des productions et leurs emplacements se trouvent sur http://shorturl.at/iuEST						

¹ En référence aux *conditions de réalisation et ressources nécessaires* du bloc « Conception et développement d'applications » prévues dans le référentiel de certification du BTS SIO.

² Les réalisations professionnelles sont élaborées dans un environnement technologique conforme à l'annexe II.E du référentiel du BTS SIO.

³ Conformément au référentiel du BTS SIO « Dans tous les cas, les candidats doivent se munir des outils et ressources techniques nécessaires au déroulement de l'épreuve. Ils sont seuls responsables de la disponibilité et de la mise en œuvre de ces outils et ressources. La circulaire nationale d'organisation précise les conditions matérielles de déroulement des interrogations et les pénalités à appliquer aux candidats qui ne se seraient pas munis des éléments nécessaires au déroulement de l'épreuve. ». Les éléments peuvent être un identifiant, un mot de passe, une adresse réticulaire (URL) d'un espace de stockage et de la présentation de l'organisation du stockage.

Descriptif de la réalisation professionnelle, y compris les productions réalisées et schémas explicatifs

Contexte :

La solution envisagée est la mise en ligne d'un site internet permettant aux utilisateurs de s'inscrire et de s'identifier, puis d'utiliser les fonctionnalités de réservation de salle, d'ajout de participant aux réunions et de création de tickets de réclamation.

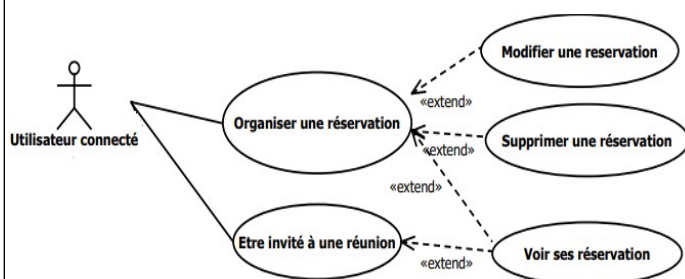
Je me suis occupé plus spécifiquement :

- Gestion d'une réservation (Ajout/suppression invités + supprimer une réservation).
- Historique d'une réservation
- Affichage des invitations (Réunions) & Historique
- Gestion du compte

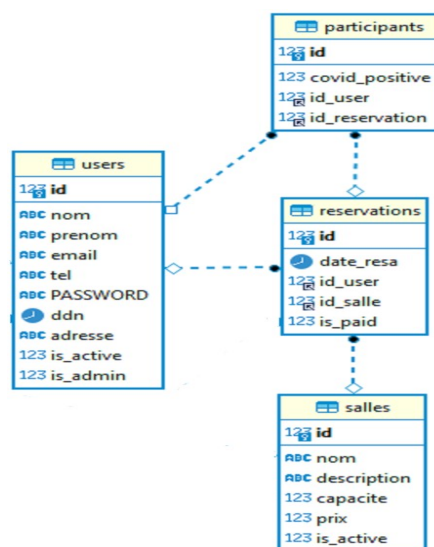
Procédure de développement(exemple) :

Développement de la fonctionnalité de Gestion d'une réservation.

1. Elaboration du Use Case diagramme



2. Définition du Modèle Logique de Données



Avec mon groupe de projet, nous avons ensuite créé le schéma de la base de données (BDD) et un jeu de données de test sous MariaDB.

Après avoir listé toutes les fonctionnalités liées à la réservation d'une salle, je les ai traduits dans le langage SQL et créés des procédures stockées dans le but d'accéder à la base de données via notre API (Interface de Programmation d'Application).

Pour la création de notre API, nous avons utilisé l'environnement Node.js et mis en place un serveur Express.js pour faciliter le développement et pour plus de sécurité. Des chemins URL ont été défini pour l'utilisation de ces procédures et ont été protégé pour que seuls les utilisateurs connectés puissent réserver une salle.

Pour finir, j'ai conçu la partie front-end permettant à l'utilisateur de gérer ses réservations, découpant ça en 2 composants, un affichant une liste des réservations à venir, un second se déployant suite à un click sur une des réservations, permettant de gérer la liste des invités de celle ci.

⁴ Lien vers la documentation complète, précisant et décrivant, si cela n'a été fait au verso de la fiche, la réalisation professionnelle, par exemple service fourni par la réalisation, interfaces utilisateurs, description des classes ou de la base de données.

3. Captures d'interface web liée à la réservation d'une salle

Liste des Réservations



Accueil Salles **Réservations** Réunions Réclamations Administration

test1_prenom test1_nom
Log Out

Vos prochaines réservations

Nom de la Salle	Description	Date	Gérer	Supprimer
Amphithéâtre	Service de sonorisation et vidéo projecteur disponible	15/04/2022	<button>Gérer</button>	<button>Annuler</button>
Gallé	Vidéo projecteur disponible	25/04/2022	<button>Gérer</button>	<button>Annuler</button>
Longwy	Vidéo projecteur disponible	01/01/2023	<button>Gérer</button>	<button>Annuler</button>

Historique des réservations

Gestion d'une Réservation



Accueil Salles **Réservations** Réunions Réclamations Administration

test1_prenom test1_nom
Log Out

Réservation de la salle Amphithéâtre à la date du 15/04/2022

Ajouter un participant

Ajouter

Revenir à mes réservations

Participants

Nom	Prénom	Email	Supprimer
test1_nom	test1_prenom	test1@email.com	
test4_nom	test4_prenom	test4@email.com	<button>Yeet</button>
test2_nom	test2_prenom	test2@email.com	<button>Yeet</button>

Test de bon fonctionnement :

Les fonctionnalités de visualisation de salle et de réservation ont été testée comme suit :

- Test de la procédure stockées SQL via le logiciel DBeaver.
- Test des points d'accès (URL) et contrôleur (Node.js), via Postman, ainsi que la méthode appelé (GET, POST, etc...).
- Test unitaire, test d'intégration et d'utilisation de l'interface.

4. Exemple de test de point d'accès avec Postman



```
GET localhost:3001/users

Query Params
KEY VALUE DESCRIPTION
Key Value Description

Body
200 OK 13 ms 1.26 KB

Pretty Raw Preview Visualize JSON
{
  "success": [
    {
      "id": 1,
      "nom": "test1_nom",
      "prenom": "test1_prenom",
      "email": "test1@email.com",
      "tel": "0123456789",
      "ddn": "1990-12-31T23:00:00.000Z",
      "adresse": "test1_adresse"
    }
  ]
}
```