

2025/2026	AP – Itération 2
BTS SIO 2SLAM	Auteur : Taillé Jade & Delaunay-Guitton Benjamin & Portolleau Anaïs Date de rédaction : 29/09/2025

Projet site web R3st0.fr - Itération n°2 – résolution de tickets

Ticket n° 7 - affichage de la valeur numérique de la note moyenne d'un restaurant

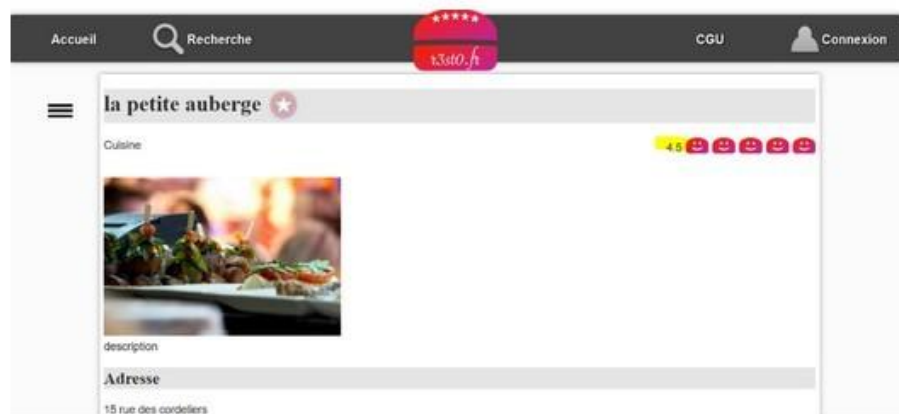
Modifier



○ Ouvert ✓ Tâche créé Il y a 13 secondes par Bourgeois

Modifier l'affichage du détail d'un restaurant en rajoutant la note moyenne **non arrondie** devant les têtes de bonhommes.

Par exemple pour le détail du restaurant La petite auberge, on aurait ceci :



🔗 Créer une requête de fusion ▼

Éléments liés 0

Ajouter



Reliez des éléments pour mettre en évidence leur relation.

Personne assignée

Modifier

Aucun(e) - assigner à vous-même

Labels

Modifier

peu complexe x

priorité normale x

Parent

Modifier

Aucun

Jalon

Modifier

Itération n°2

Dates

Modifier

Début: 22 sept. 2025

Échu: 29 sept. 2025

Suivi du temps



Ajoutez une estimation ou le temps passé.

1 participant



2025/2026	AP – Itération 2
BTS SIO	Auteur : Taillé Jade & Delaunay-Guitton Benjamin & Portolleau Anaïs
2SLAM	Date de rédaction : 29/09/2025

Projet site web R3st0.fr - Itération n°2 – résolution de tickets

J'ai d'abord regardé s'il n'y avait pas de méthode pour afficher la note moyenne d'un restaurant et j'ai trouvé cette méthode qui serait parfaite ducoup je vais l'utiliser qui se situe dans le fichier modele/dao/CritiqueDao.class.php :

```
/**
 * Note moyenne de toutes les critiques pour un restaurant donné
 * @param int $idR identifiant du restaurant
 * @return float note moyenne ou -1 si aucune critique
 * @throws Exception Transmission des erreurs PDO éventuelles
 */
public static function getNoteMoyenneByIdR(int $idR): float {
    $retour = -1;
    try {
        $requete = "select avg(note) as moyenne from critiquer where idR=:idR";
        $stmt = Bdd::getConnexion()->prepare($requete);
        $stmt->bindParam(':idR', $idR, PDO::PARAM_INT);
        $ok = $stmt->execute();
        if ($ok && $stmt->rowCount() > 0) {
            $enreg = $stmt->fetch(PDO::FETCH_ASSOC);
            // Même si la valeur de $idR n'apparaît dans aucun enregistrement, la requête retourne NULL dans le champ 'moyenne' !
            if (!is_null($enreg["moyenne"])) {
                $retour = $enreg["moyenne"];
            }
        }
    } catch (PDOException $e) {
        throw new Exception("Erreur dans la méthode " . get_called_class() . "::getNoteMoyenneByIdR : <br/>" . $e->getMessage());
    }
    return $retour;
}
```

La méthode getNoteMoyenneByIdR calcule la moyenne des notes pour un restaurant donné. Elle initialise \$retour = -1 si aucune critique n'existe. La requête SQL SELECT avg(note) AS moyenne FROM critiquer WHERE idR=:idR retourne NULL si pas de critiques, sinon la valeur moyenne. La méthode récupère cette valeur et la renvoie en float, ou -1 si aucune donnée. En cas d'erreur PDO, elle lève une exception avec le message correspondant.

J'ai cherché ensuite l'endroit dans la vue des détails d'un resto donc dans le fichier vue/vueDetailResto.php, l'endroit où il y aurait la note et modifié et mit ceci:

2025/2026	AP – Itération 2
BTS SIO 2SLAM	Auteur : Taillé Jade & Delaunay-Guitton Benjamin & Portolleau Anaïs Date de rédaction : 29/09/2025

Projet site web R3st0.fr - Itération n°2 – résolution de tickets

```

37
38 <span id="note">
39     <?php if (isset($noteAffichee)) { ?>
40         <span class="note-numerique"><?= $noteAffichee ?></span>
41     <?php } ?>
42
43     <?php for ($i = 1; $i <= 5; $i++) { ?>
44         <a class="aimer" href="./?action=noter&note=<?= $i ?>&idR=<?= $unResto->getIdR(); ?>" >
45             <?php if ($i <= $noteMoy) { ?>
46                 
47             <?php } else { ?>
48                 
49             <?php } ?>
50         </a>
51     <?php } ?>
52 </span>

```

Dans le code original, le `` affichait uniquement les 5 bonhommes en fonction de la note arrondie (`$noteMoy`), sans afficher la valeur numérique réelle de la moyenne. Le changement consiste à ajouter la note moyenne tronquée à 2 décimales devant les bonhommes. On utilise `$noteAffichee`, calculée dans le contrôleur en tronquant `$noteMoyBrut`. Ensuite, la boucle des 5 bonhommes utilise toujours `$noteMoy` pour afficher le nombre de bonhommes pleins et le reste en bonhommes vides.

Et j'ai modifié par ceci :

2025/2026	AP – Itération 2
BTS SIO	Auteur : Taillé Jade & Delaunay-Guitton Benjamin & Portolleau Anaïs
2SLAM	Date de rédaction : 29/09/2025

Projet site web R3st0.fr - Itération n°2 – résolution de tickets

```
// Note moyenne brute (peut être -1 si aucune critique)
$noteMoyBrut = CritiqueDAO::getNoteMoyenneByIdR($idR);

if ($noteMoyBrut >= 0) {
    // Tronquer à 2 décimales (non arrondi)
    $noteMoyTronquee = floor($noteMoyBrut * 100) / 100;
    $noteAffichee = number_format($noteMoyTronquee, 2, '.', '');
    // Pour les bonhommes : arrondi à l'entier le plus proche
    $noteMoy = round($noteMoyBrut);
} else {
    $noteAffichee = "—";
    $noteMoy = 0;
}
```

getNoteMoyenneByIdR te donne la moyenne brute, par exemple 4.367. \$noteAffichee contient la moyenne tronquée à 2 décimales pour l’affichage, par exemple 4.36. \$noteMoy est la moyenne arrondie à l’entier pour afficher le nombre de bonhommes pleins, par exemple 4. Dans vueDetailResto.php, on place <?= \$noteAffichee ?> juste avant la boucle des bonhommes. Le résultat final est par exemple : 4.36