**Rapport Logiciel au niveau des salles**

Réaliser par : Antoine Girard étudiant au Lycée Saint Michel en BTS SIO Option SLAM

Sommaire

Sommaire

[Présentation 2](#_Toc71280446)

[Logiciel à fournir 3](#_Toc71280447)

[Logiciel Utilisé 4](#_Toc71280448)

[Langages Utilisé 5](#_Toc71280449)

[Diagramme de cas pour le logiciel du Technicien 6](#_Toc71280450)

[Scénario use case 7](#_Toc71280451)

[Diagramme de classes 8](#_Toc71280452)

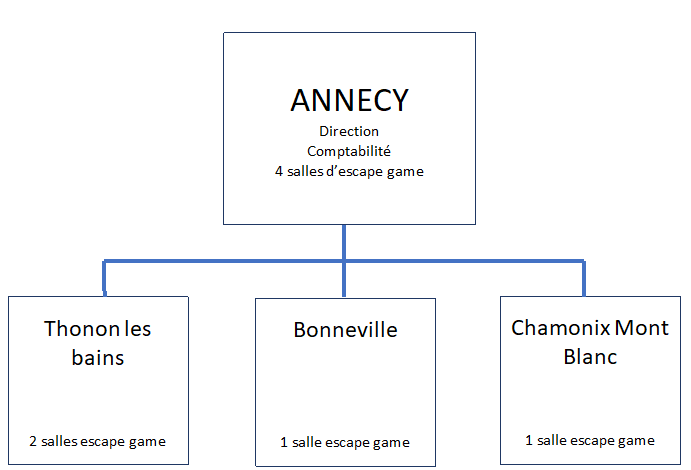
[Cahier de recette 9](#_Toc71280453)

[Guide d’installation logiciel 10](#_Toc71280454)

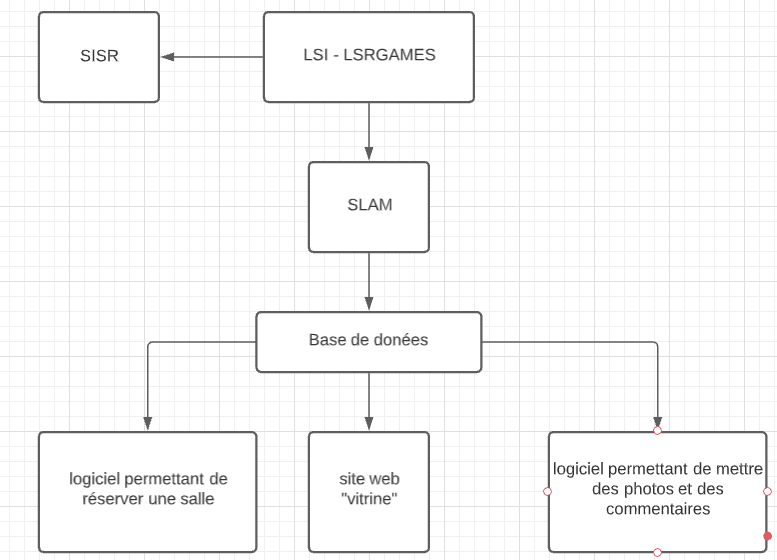
# Présentation

La EGC(Escape Game Corporation) est une nouvelle entreprise gérant des salles d’escape game dans le département de la Haute-Savoie. Elle est sous statut SCOP.

Elle disposera de 4 sites de jeu situé à Annecy, Thonon les bains, Chamonix Mont-Blanc et enfin Bonneville. Ces villes ont été choisies par rapport à leur potentiel en termes de joueurs. Le siège social sera basé à Annecy.



# Logiciel à fournir



Base de données : la base de données sera réalisée avec MySQL.

Logiciel de réservation de salle : il permet d’observer ses résultats et de réserver une salle

Site web “vitrine” : qui affiche des photos aléatoirement, des commentaires. Il donne aussi des informations en temps réel.

Logiciel de prise de photos et de commentaires : il permet de mettre des photos, des commentaires à chaud.

Je suis la personne qui va concevoir le logiciel de prise de photos et de commentaires, donc nous allons plus nous pencher sur ce logiciel que sur les autres logiciels.

# Logiciel Utilisé

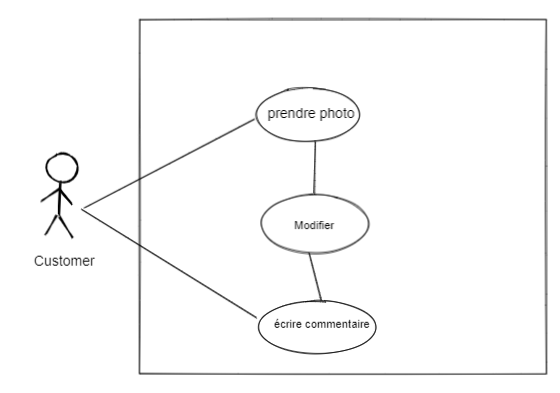
|  |  |  |
| --- | --- | --- |
| Visual Studio Code — Wikipédia | MySQL — Wikipédia | Symfony logo and screenshots |
| Visual Studio Code | MYSQL | Symfony |

|  |  |
| --- | --- |
| API Platform: REST and GraphQL framework on top of Symfony and React | GitHub - t4t5/sweetalert: A beautiful replacement for JavaScript's "alert" |
| Api platform (création api) | Sweet Alert js (pop up personnalisé) |

# Langages Utilisé

|  |  |  |  |
| --- | --- | --- | --- |
| Logo Html Html5 - Image gratuite sur Pixabay | Feuilles de style en cascade — Wikipédia | File:JavaScript-logo.png - Wikimedia Commons | Fichier:PHP-logo.svg — Wikipédia |
| HTML | CSS | JavaScript | PHP |

# Diagramme de cas pour le logiciel du Technicien



# Scénario use case

**Prendre photo** : le client peut prendre des photos.

**Écrire commentaire** : le client peut écrire un commentaire.

**Modifier** : pendant la connexion il aura le choix de sois supprimer son commentaire mais aussi de supprimer les photos.

**Scénario principal :**

1-le client prend une photo

2-l'application lui demande s’il veut sauvegarder l’image

3-le client accepte

4- le client termine sa partie et donc appuie sur le bouton terminé

5-le client rentre le commentaire

6-le client rentre une note

7-l'application lui propose de sauvegarder son commentaire et ces images

8-le client accepte

9-le commentaire et les images sont enregistrés

**Scénario alternatif :**

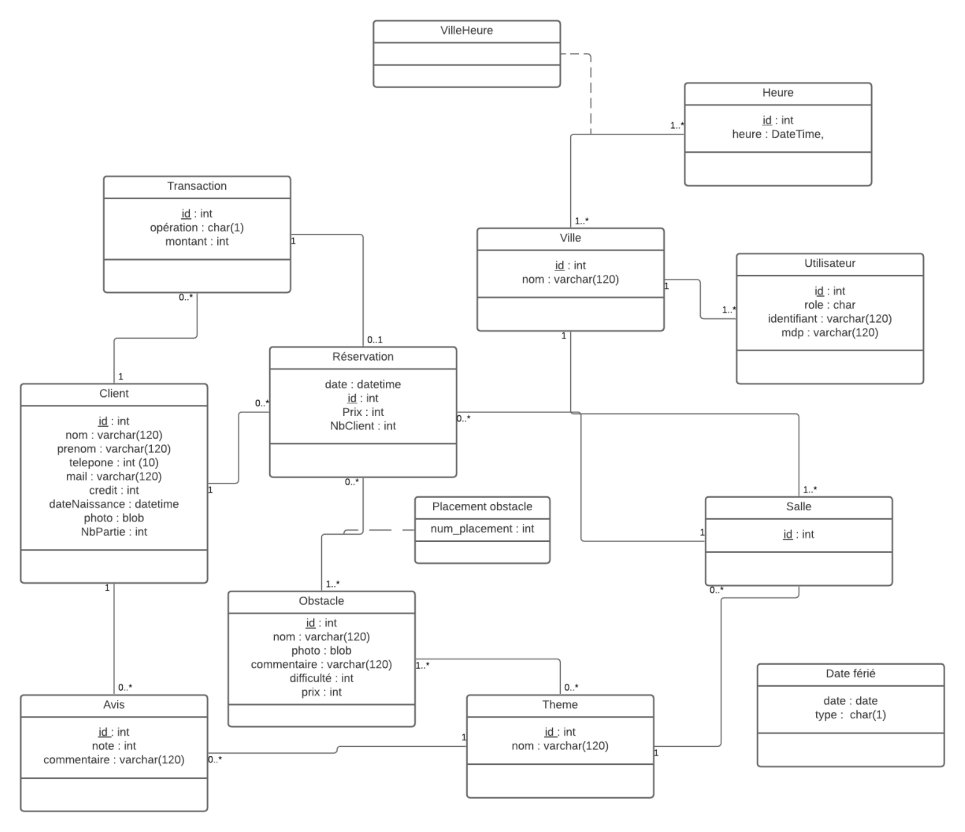
3.1 - le client refuse

3.2 - l’image est supprimée

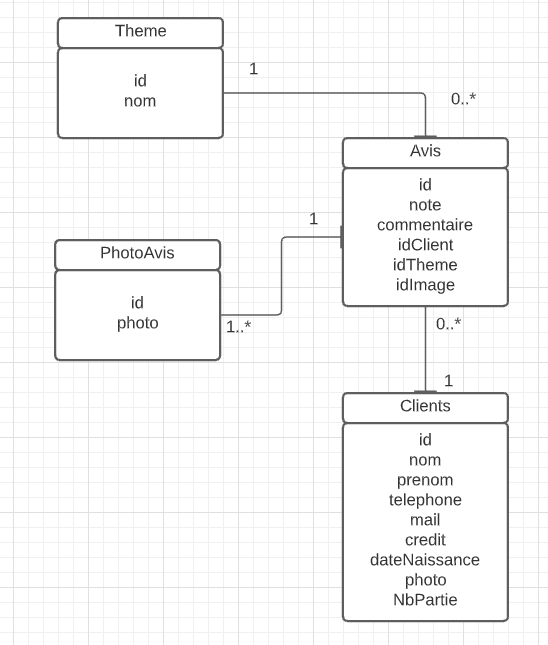
8.1 - le client refuse

8.2 - l’image et le commentaire sont supprimés

# Diagramme de classes



Seul les classes : Obstacle, Client, Theme, Avis et une nouvelle classe PhotoAvis seront utiliser.



# Cahier de recette

|  |  |  |  |
| --- | --- | --- | --- |
| N° | Action | Attendu | Résultat |
| **Fonctionnement général de l’application** | | | |
| 1 | Prendre une photo | Prends une photo |  |
| 2 | Ajouter un commentaire | Enregistre le commentaire |  |
|  | Terminer la partie | Enregistre dans la base de données la photo et le commentaire |  |

# Guide d’installation logiciel

Sur mon github est disponible dans l’url : https://github.com/AntoineGir/ppe4\_symfony

Le code du logiciel.

Après l’avoir téléchargé, aller dans le dossier avec le cmd et taper cette commande :

**“composer update”**

Vous devez aussi changer la base de données, le nom et le mot de passe pour faire ceci

Aller dans le fichier .env et modifier la ligne “DATABASE\_URL”.

Après ceci il faut créer la base de données : **“php bin/console doctrine :database :create”**

Après ceci taper cette commande **“php bin/console make:migration”**

Puis : **“php bin/console doctrine :migrations :migrate”**

Pour lancer l’application tapez : **“symfony serve –d”**

Puis tapez dans url de votre navigateur**:** **“https://127.0.0.1:8000/home”**