



Site et application mobile GILLA

Documentation technique

Sommaire

| | |
|---|----|
| Cahier des Charges | 2 |
| Organisation du projet GILLA | 6 |
| Spécifications fonctionnelles..... | 12 |
| Spécifications techniques | 29 |
| Conception de la solution | 31 |
| Codage du composant com_gilla..... | 39 |
| Application AndroidGilla | 45 |
| Intégration et tests d'intégration du site GILLA..... | 48 |
| Tests et recette fonctionnelle | 51 |
| Déploiement..... | 54 |

Cahier des Charges

Présentation du Lycée Louis Armand

Le lycée polyvalent Louis Armand situé à Paris 15^e offre aux élèves de seconde générale un parcours diversifié :

- Bac général : Bac Spé Maths/PC/NSI/SES/HG/SVT.
- Bac technologique : Bac STI2D et Bac STMG.
- Bac professionnel : Bac Pro MELEC / Bac Pro CIEL /Bac Pro Photographie.
- CAP électricien.
- UPEAA.

Après le BAC, le lycée offre les formations supérieures suivantes :

- BTS CIEL (anc. Systèmes Numériques).
- BTS Gestion de la PME.
- BTS Management Commercial Opérationnel.
- BTS Services Informatiques aux Organisations.
- BTS Assistant Technique d'Ingénieur
- BTS Photographie en alternance.
- BTS Assistant Technique d'Ingénieur en alternance.

Objectif du projet GILLA

L'objectif du projet GILLA est de mettre en place un site permettant aux élèves, étudiants, professeurs et personnels du lycée de signaler des problèmes de tout type (éclairage, fermeture porte ou fenêtre, matériel informatique ou logiciel, etc.) et prendre en charge leur résolution.

Le site GILLA doit permettre une gestion centralisée des « tickets d'incidents » et une affectation automatique aux personnels de maintenance selon le type d'incident identifié, pour une prise en charge rapide.

Analyse de l'existant

La gestion des incidents au lycée se fait actuellement de façon désordonnée par email ou oralement à divers personnels responsables au lycée. Les incidents ne sont pas gérés de façon centralisée pour pouvoir les traiter efficacement en fonction des ressources de maintenance disponibles, selon leurs expertises.

Une application gérée par la région permet de signaler des problèmes informatiques qui sont traités toutes les deux ou trois semaines par un intervenant technique de la région. Elle est en pratique très peu utilisée au lycée.

L'organisation du personnel du lycée est donnée en page suivante.

Organigramme des services

| | |
|---|--|
| Service direction : | Proviseur : Mme G.Malleville Proviseur Adjoint : Mme C. Ali-Cherif |
| Service intendance : <i>Cantine / Bourses</i> | Intendant : L. Cherfaoui Secrétariat d'intendance : R. Vairac Secrétariat d'intendance : W.Bancoul |
| Service DDFPT : <i>Stages / PFMP / Ateliers / Partenariats professionnels</i> | DDFPT : G. Idres Assistante DDFPT : M-P Lebreton |
| Service vie scolaire : <i>AED / Suivi scolaire</i> | CPE : H. Horatius CPE : S. Fratta CPE : C. Zannier <i>Quelle CPE pour quelle classe ici</i> |
| Pôle médico social : <i>Santé / Orientation</i> | Infirmière : N. Bourreau Médecin scolaire : C. Visseriat Assistante sociale : P. Jeay PsyEn : Mme O.Descout |
| Pôle maintenance et entretien | Responsable : M. Gesbert |

L'équipe de maintenance comprend les quatre agents suivants :

- M. Gualbert GESBERT : Responsable
- M. El Hassane BELLAOUI : Agent de maintenance
- M. Tarek BRIDIA : Agent de maintenance
- M. Manuel DA-CUNHA : Agent de maintenance

Ils prennent en charge l'entretien des éléments des trois types suivants :

- Bâtiment : portes, fenêtres, murs et sols.
- Mobilier : tables, chaises et armoires.
- Informatique : vidéoprojecteurs, ordinateurs, imprimantes.

Chaque élément est situé dans un emplacement de type salle, local, couloir ou escalier.

Le plan du lycée permet d'identifier les salles par étage sous la forme XYnn, X étant la lettre du bâtiment (A, B, C, D ou F), Y le numéro de l'étage et nn étant les deux chiffres du numéro de salle (ex : A313 : bâtiment A, étage 3, numéro 13).

D'un point de vue technique, l'infrastructure technique du lycée dispose :

- D'un réseau Wifi avec des bornes dans chaque salle et comportant plusieurs réseaux privés : LARM élèves, LARM Professeur, etc.
- D'un réseau local filaire (LAN) Ethernet commuté, accessible depuis toutes les salles par prises RJ45 et configuré en plusieurs VLAN.
- D'un serveur proxy académique filtrant les accès Internet et accessible par le réseau Wifi et le réseau local.
- Un serveur de domaine « LouisArmand » comprenant un annuaire LDAP pour gérer les autorisations de connexion de tous les utilisateurs du lycée et un serveur de fichier.
- Des serveurs réservés à certains groupes d'utilisateurs (Administration, BTS-SIO...), accessibles à certains VLAN.

Analyse des besoins

A l'issue d'une série d'entretiens réalisés auprès du directeur technique du lycée, M. IDRES, les besoins généraux de conception du système de gestion des incidents au lycée Louis Armand se résument comme suit :

- Faciliter le signalement d'incidents au lycée, quelque-soient leurs types.
- Rendre accessible le signalement d'incidents à toute personne travaillant au lycée.
- Centraliser l'enregistrement des incidents et leurs affectations aux personnels de maintenance.
- Accélérer la prise en charge des incidents et les opérations de maintenance correspondantes.
- In fine, favoriser une amélioration continue de la qualité des infrastructures du lycée.

Les besoins spécifiques du site GILLA sont notamment les suivants :

- Permettre à toute **personne** travaillant au lycée (élève, étudiant, enseignant, AED, CPE, personnel administratif, agent technique) de signaler un incident et de préciser le type d'élément concerné (par sélection dans une liste) et son emplacement avec un commentaire et une photo si besoin ; lui permettre aussi de suivre la prise en charge des incidents qu'il a signalé.
- Permettre à un **agent** de maintenance de consulter les incidents du ou des type(s) dont il a la charge, de gérer leurs priorités (Haute, Moyenne ou Basse) et leurs états d'avancement (Affecté, Résolu ou Fermé) et de les clôturer avec un commentaire.
- Permettre au **responsable** de la maintenance de gérer l'équipe de maintenance et les affectations des agents aux types d'incidents ; lui permettre aussi de disposer d'un tableau de bord pour suivre la prise en charge des incidents par type et par personnel de maintenance.
- Permettre à un **administrateur** du site de gérer les profils des utilisateurs inscrits et leurs rôles associés (groupe d'utilisateur) ; lui permettre aussi de mettre à jour le site et la base de données associée.

Une analyse détaillée de ces besoins à partir d'interview des principaux utilisateurs concernés devra permettre d'établir un document de spécifications fonctionnelles à faire valider par les parties prenantes du projet en charge de sa coordination, avant de lancer les phases d'exécution technique du projet.

D'un point de vue technique, le site GILLA devra être accessible depuis tout navigateur Internet et être « *responsive* », c'est-à-dire adapté à tout type d'écran : PC/Mac, tablette ou téléphone

(smartphone).

Le prestataire (MOE) devra concevoir le site à partir d'un framework PHP (Joomla 4 ou Symfony 5) pour sécuriser la solution et faciliter son développement et sa maintenance.

Une application mobile (Android dans un premier temps) sera aussi proposée aux utilisateurs pour signaler des incidents et suivre leurs prises en charge depuis leur téléphone. Celle-ci devra communiquer via une API sécurisée à la base de données du site.

Tout au long de son déroulement, ce projet de développement logiciel devra être soigneusement documenté dans ses aspects fonctionnels et techniques afin de faciliter la communication entre la MOA et la MOE.

Organisation du projet GILLA

Organisation générale du projet

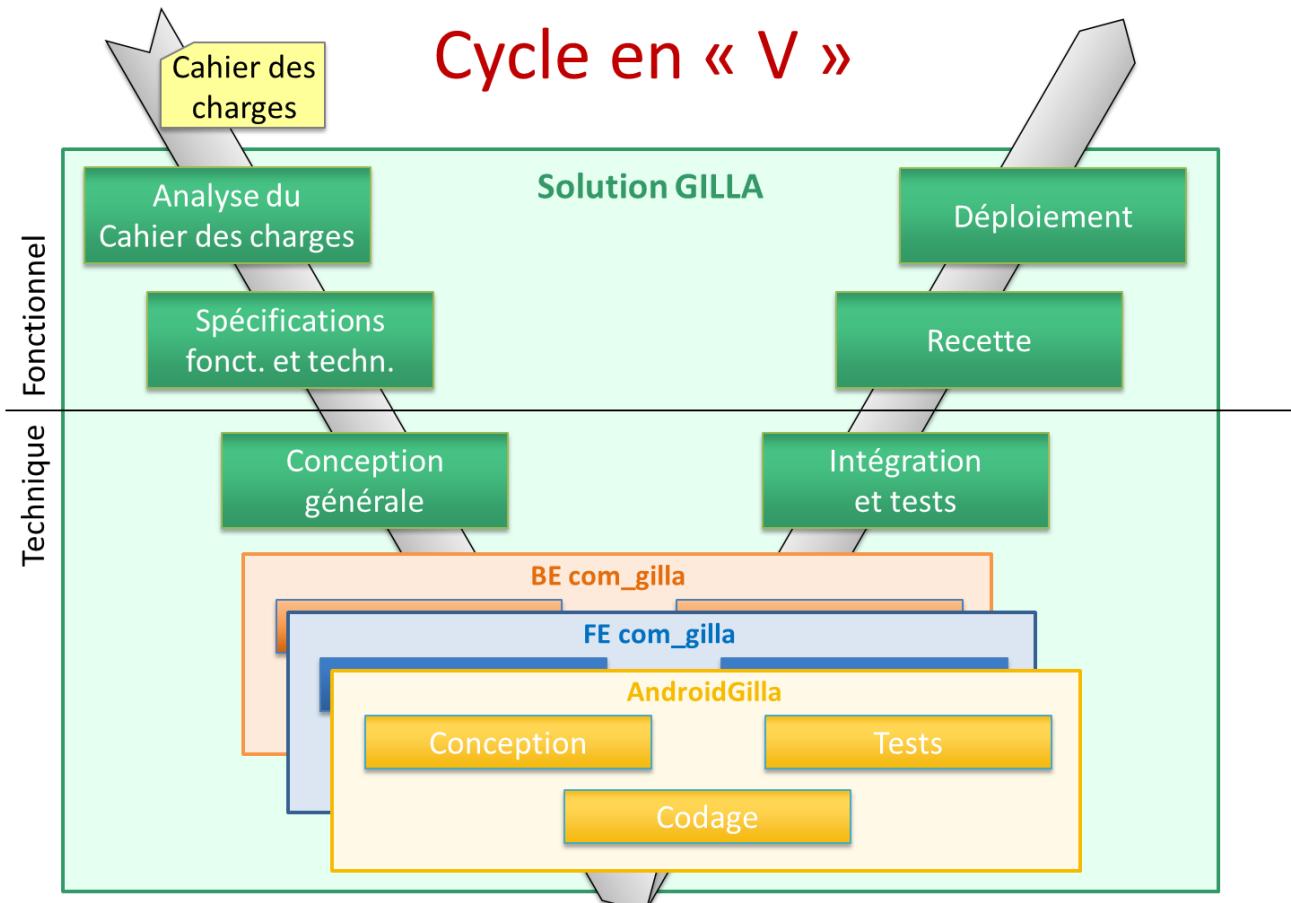
Le développement du projet GILLA a été confié aux étudiants de deuxième année de BTS SIO option SLAM du lycée. L'organisation du projet mise en place par le professeur en charge du bloc 2 du référentiel du BTS SIO, a coïncidé avec la progression pédagogique établie pour ce bloc, le projet GILLA servant de fil conducteur au déroulé des cours et des travaux pratiques.

Le projet GILLA a d'abord été décomposé en trois livrables principaux liés à l'architecture du framework Joomla retenu et présenté plus loin (cf. chapitre conception et architecture) :

- Le **site d'administration**, avec le développement de la partie backend du composant com_gilla (extension Joomla spécialement développée pour le projet).
- Le **site public** incluant l'accès réservé aux utilisateurs enregistrés avec la configuration du CMS Joomla et le développement de la partie frontend du composant com_gilla.
- Le développement de l'**application mobile Android** et de l'API associée.

Approche méthodologique

Le développement s'est fait selon la méthodologie du cycle de vie des logiciels (cycle en « V ») définissant les grandes étapes du projet depuis l'analyse du cahier des charges jusqu'à la recette fonctionnelle et au déploiement :



Le cycle en « V » de la solution GILLA commence à partir du cahier des charges, avec les trois étapes suivantes :

- Analyse du cahier des charges.
- Spécifications fonctionnelles et techniques.
- Conception générale.

A partir de cette étape, l'architecture générale de la solution est définie et trois développements sont identifiés :

1. Le backend du composant com_gilla.
2. Le frontend du composant com_gilla.
3. L'application mobile AndroidGilla.

Ces trois développements se décomposent en trois sous-cycles en « V » :

- Conception
- Codage
- Test

Une fois ces trois développements terminés, le cycle en « V » principal continue, avec les trois étapes suivantes :

- Intégration et tests.
- Recette.
- Déploiement.

Le déploiement aboutit dans ce projet école à la mise en production d'une solution Web et mobile prototype, sur des serveurs de l'hébergeur OVH.

Planning général

Le planning général du projet GILLA et de ses trois sous-projets (BE : Backend, FE : Frontend, AM : Application Mobile) s'est inscrit dans le calendrier (progression pédagogique) mis en place sur l'année scolaire 2023-2024 comme suit :

| Semaine de cours | | | Thème | Séquence |
|-----------------------|-------|-------|-----------------------|--|
| N° | Lu | Sa | | |
| 0 | 28/8 | 2/9 | | |
| 1 | 4/9 | 9/9 | Analyser la demande | 1 - Analyse du cahier des charges de la solution |
| 2 | 11/9 | 16/9 | | |
| 3 | 18/9 | 23/9 | | |
| 4 | 25/9 | 30/9 | Spécifier la solution | 2 - Spécifications technique et fonctionnelle de la solution |
| 5 | 2/10 | 7/10 | | |
| 6 | 9/10 | 14/10 | Concevoir la solution | 3 - Architecture de la solution |
| 7 | 16/10 | 21/10 | | |
| Vac. Toussaint | | | | |
| 8 | 6/11 | 11/11 | Coder le BE | 5a - Codage du BE du composant |
| 9 | 13/11 | 18/11 | Tester le BE | 6a - Tests du BE du composant |
| 10 | 20/11 | 25/11 | Concevoir le FE | 4b - Architecture du FE du composant |
| 11 | 27/11 | 2/12 | Coder le FE | 5b - Codage du FE du composant |
| 12 | 4/12 | 9/12 | | |
| 13 | 11/12 | 16/12 | Tester le FE | 6b - Tests du FE du composant |
| 14 | 18/12 | 23/12 | | Stage en entreprise (6 semaines) |
| Vac. Noël | | | | |
| 15 | 8/1 | 13/1 | | |
| 16 | 15/1 | 20/1 | | |
| 17 | 22/1 | 27/1 | | |
| 18 | 29/1 | 3/2 | | |
| 19 | 5/2 | 10/2 | | |
| Vac. Hiver | | | | |
| 20 | 26/2 | 2/3 | Concevoir l'AM | 4c - Architecture de l'application mobile |
| 21 | 4/3 | 9/3 | Coder l'AM | 5c - Codage de l'application mobile |
| 22 | 11/3 | 16/3 | | |
| 23 | 18/3 | 23/3 | Tester l'AM | 6c - Tests de l'application mobile |
| 24 | 25/3 | 30/3 | Documenter | 8 - Recette et documentation de la solution |
| 25 | 1/4 | 6/4 | | CCF E5 |
| Vac. Printemps | | | | |

Gestion du projet

Des groupes de travail ont été constitués pour faciliter le partage de connaissance et la collaboration entre les étudiants mis en situation professionnelle lors des travaux pratiques. Des contributions individuelles au projet ont été aussi demandées à chaque étudiant pour permettre à chacun de développer son autonomie.

Les tableaux suivants identifient, les **responsables des développements par acteur et par rubrique**, pour les groupes A et B.

Groupe A :

Backend de com_gilla :

| Acteur | Rubrique | Vue liste | | Vue détail | |
|----------------|------------------|-----------|---------|------------|-------------|
| Administrateur | Affectations | RDP | Aymen | CRU | Gustave |
| | Agents | RDP | Matéo | CRU | Hugo |
| | Emplacements | RDP | Davy | CRU | Radu |
| | Etats | RDP | Ilan | CRU | |
| | Incidents | RDP | Ryan | CRU | Mathis |
| | Priorités | RDP | | CRU | |
| | Prises en charge | RDP | Malik | CRU | Abdel-Karim |
| | Types | RDP | Bayrone | CRU | Oumar |

Frontend de com_gilla :

| Acteur | Rubrique | Vue liste | | Vue détail | |
|-------------|----------------------|-----------|-------------|------------|---------|
| Utilisateur | Incidents ouverts | R | Hugo | R | Aymen |
| | Signaler un incident | | | CRU | Davy |
| | Mes incidents | R | Gustave | R | Malik |
| Agent | Incidents à gérer | R | Radu | R | Matéo |
| | Prises en charge | R | Mathis | CRU | Ilan |
| Responsable | Agents | R | Oumar | CRU | Ryan |
| | Affectations | R | Abdel-Karim | CRU | Bayrone |

AndroidGilla :

| Acteur | Rubrique | Vue liste | | Vue détail | |
|-------------|----------------------|-----------|--------|------------|--------|
| Utilisateur | Incidents ouverts | R | (tous) | R | (tous) |
| | Signaler un incident | | | CRU | (tous) |
| | Mes incidents | R | | R | |
| Agent | Incidents à gérer | R | | R | |
| | Prises en charge | R | | CRU | |

Groupe B :**Backend de com_gilla :**

| Acteur | Rubrique | Vue liste | | Vue détail | |
|----------------|------------------|-----------|------------|------------|---------------|
| Administrateur | Affectations | RDP | Agahlya | CRU | Aïssata |
| | Agents | RDP | Abdenour | CRU | Alexy |
| | Emplacements | RDP | Tyron | CRU | Yanis |
| | Etats | RDP | Gabriel F. | CRU | Gabriel F. |
| | Incidents | RDP | Gabriel P. | CRU | Omar |
| | Priorités | RDP | | CRU | |
| | Prises en charge | RDP | Melvyn | CRU | Julian |
| | Types | RDP | Clément | CRU | Jean-François |

Frontend de com_gilla :

| Acteur | Rubrique | Vue liste | | Vue détail | |
|-------------|----------------------|-----------|------------|------------|---------------|
| Utilisateur | Incidents ouverts | R | Omar | R | Jean-François |
| | Signaler un incident | | | CRU | Melvyn |
| | Mes incidents | R | Aïssata | R | Agahlya |
| Agent | Incidents à gérer | R | Yanis | R | Clément |
| | Prises en charge | R | Gabriel F. | CRU | Abdenour |
| Responsable | Agents | R | Alexy | CRU | Gabriel P. |
| | Affectations | R | Julian | CRU | Tyron |

AndroidGilla :

| Acteur | Rubrique | Vue liste | | Vue détail | |
|-------------|----------------------|-----------|--------|------------|--------|
| Utilisateur | Incidents ouverts | R | (tous) | R | (tous) |
| | Signaler un incident | | | CRU | (tous) |
| | Mes incidents | R | | R | |
| Agent | Incidents à gérer | R | | R | |
| | Prises en charge | R | | CRU | |

Gestion des versions logicielles

La gestion des versions logicielles est faite avec le logiciel libre **Git** qui permet :

- D'enregistrer des **états de la base de code** appelés *commits* (versions) associés à des commentaires, permettant de suivre plus facilement l'évolution du projet et de revenir en arrière si besoin.
- D'organiser les versions dans des **branches**, permettant de développer une version sans impacter les autres.
- De **travailler à plusieurs** sur le même projet, grâce à un système de merge (fusion).
- De disposer de **sauvegardes** du projet grâce à une logique de distribution. Car chaque ordinateur dispose du projet dans son intégralité.

Le système d'identification des utilisateurs de Git fonctionne avec une simple adresse email. Chaque utilisateur peut créer des *commits* à chaque fonctionnalité développée.

Pour simplifier le travail à plusieurs, un dépôt privé accessible en ligne est également créé sur le site **github.com**.

Chaque utilisateur peut ainsi envoyer ses *commits* sur github et fusionner son code avec celui des autres.

Une seule branche a été créée, la branche **master**, afin de simplifier l'intégration du code (*merge*) au fil de l'eau. Le nombre de conflits de *merge* a été limité par le fait que chaque développeur travaille sur des fichiers différents, à quelques exceptions près.

Spécifications fonctionnelles

Le site GILLA se décompose en deux parties :

- Le **backend GILLA**, site d'administration du site GILLA.
- Le **frontend GILLA**, site GILLA constitué d'une partie publique et d'une partie à accès réservé aux utilisateurs enregistrés.

Les présentes spécifications fonctionnelles définissent les acteurs, les diagrammes de cas d'utilisation et les diagrammes de séquence associés, les interfaces utilisateurs et le modèle des données commun aux deux sites.

Acteurs

Au sens UML, les acteurs concernés par le système GILLA sont les suivants :

- Backend GILLA: administrateur.
- Frontend GILLA: utilisateur, agent et responsable.

Diagramme de cas d'utilisation du backend GILLA

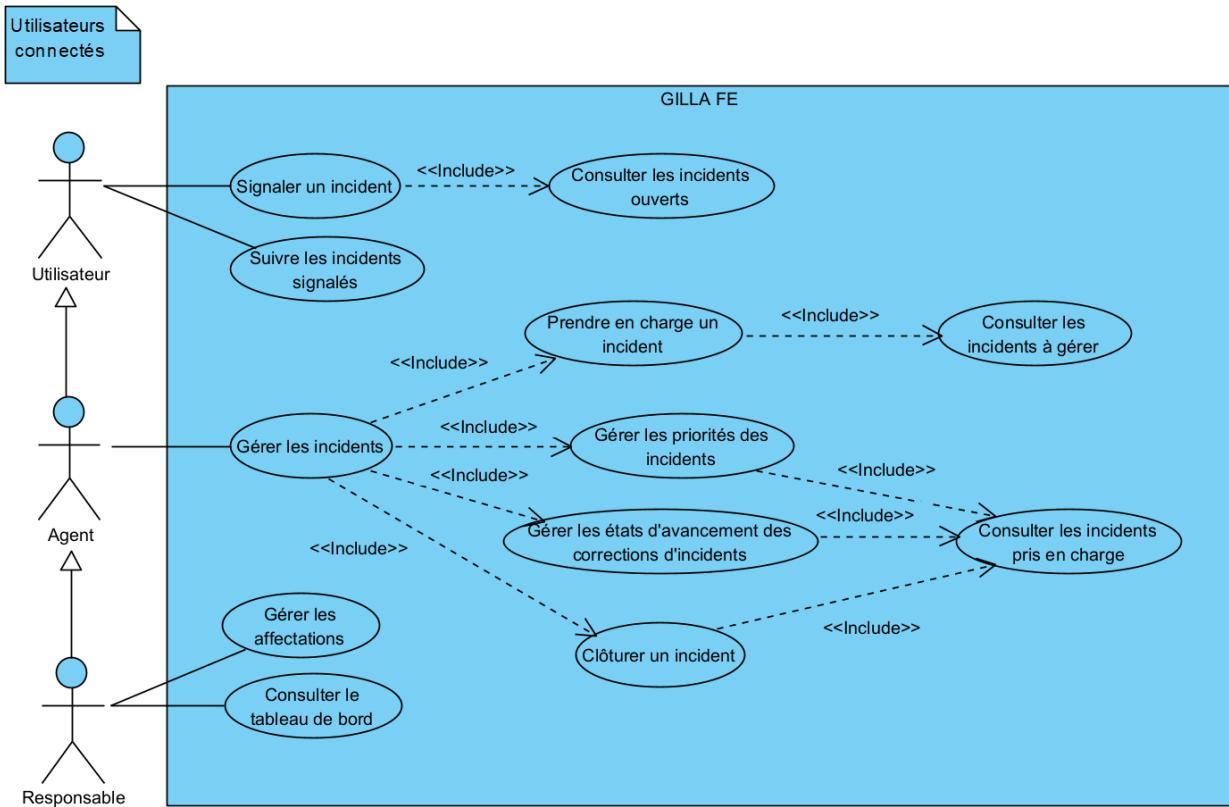
Le diagramme des cas d'utilisation UML du backend GILLA est le suivant :
(#01) Gustave



« Gérer... » inclut les 5 cas d'utilisation : Ajouter, Lire, Modifier, Supprimer et Publier/Dépublier.

Diagramme de cas d'utilisation du frontend GILLA

Le diagramme des cas d'utilisation du frontend GILLA est le suivant :



Utilisateur / Signaler un incident :

(#02) Davy

Description

Le formulaire de signalement d'incident est destiné à toutes personnes travaillant au lycée.

L'utilisateur est déjà connecté, il sélectionne la rubrique « signaler un incident » dans le menu après connexion au site GILLA pour voir s'afficher le formulaire de signalement d'incident.

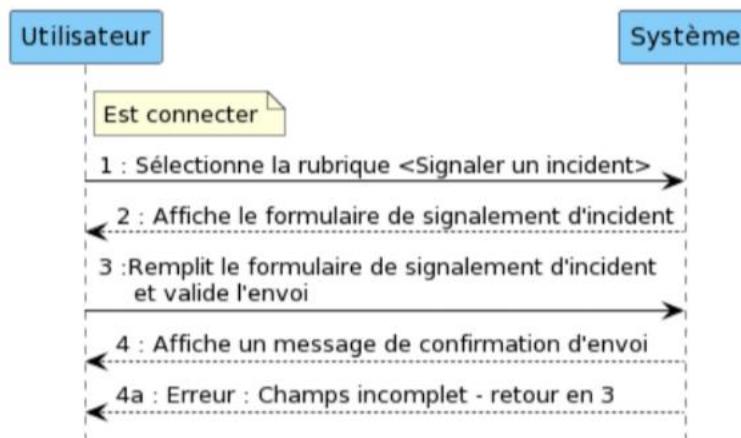
Il remplit le formulaire de signalement d'incident (Le type d'élément concerné, l'emplacement de l'incident, la description de l'incident et si nécessaire une image) et le valide.

Un message confirmant l'envoi du signalement s'affiche après la validation de l'envoi.

Maquette IHM

| GILLA | Exemple |
|------------------------------------|---|
| Signalement d'incident | |
| Type d'élément concerné: ... | Type d'élément concerné: écran |
| Emplacement de l'incident: ... | Emplacement de l'incident : Salle A313 |
| Description du de l'incident : ... | Description du de l'incident : L'écran du poste A313-P01 ne s'allume plus. |
| ^ Importer une image | ^ Importer une image |
| |  |
| ✓ Envoyer le signalement | ✓ Envoyer le signalement |

Signaler un incident

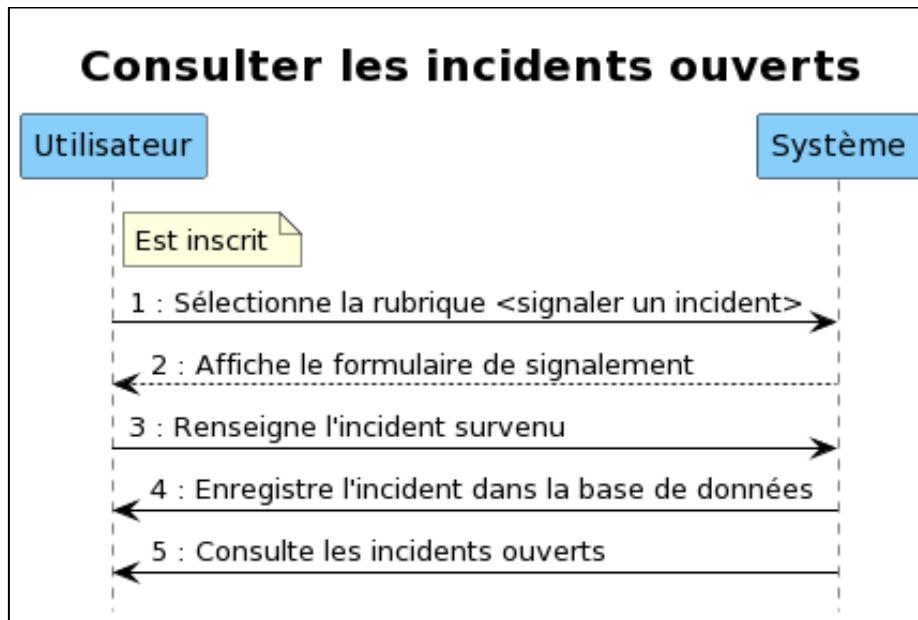


Utilisateur / Consulter les incidents ouverts :

(#03) Radu

Description

Dès lors qu'un utilisateur de l'application remarque un incident survenu, il le signale alors sur l'application. A la suite de cela, les incidents sont enregistrés dans la base de données de l'application. Je peux alors consulter l'intégralité des incidents ouverts



Agent / Prendre en charge un incident :(#04) *Malik***Description**

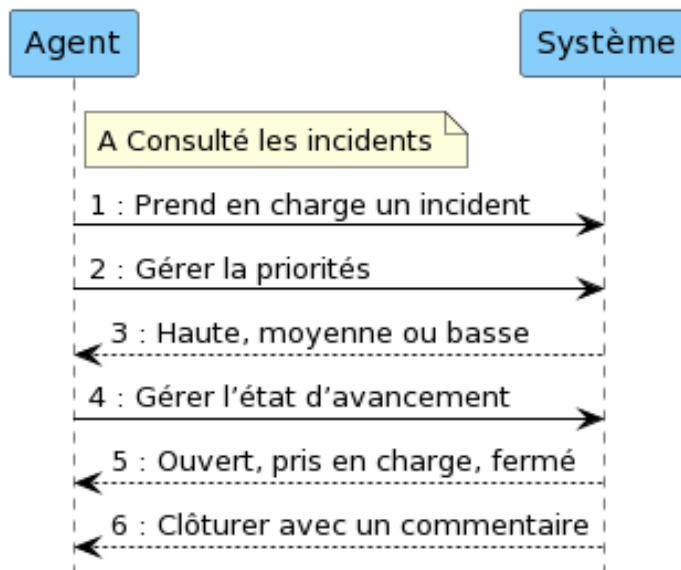
L'agent a déjà consulté les incidents à gérer du site GILLA, il a sélectionné un incident pour le prendre en charge.

Différentes options s'offrent désormais à lui :

- Il peut gérer la priorité (haute, moyenne ou basse) des incidents qu'il a pris en charge.
- Gérer l'état d'avancement (ouvert, pris en charge, fermé)
- Clôturer avec un commentaire

| GILLA | | | Priorité s | Gérer l'état | Clôturer avec commentaire |
|--------------------------|-----------------------|--|---------------------|------------------------------|----------------------------------|
| Nom de l'incident | Incident | | <i>Haute</i> | <i>Ouvert</i> | |
| Utilisateur | <i>Loulou</i> | | <i>Moyenne</i> | <u><i>Pris en charge</i></u> | |
| Description | <i>description</i> | | <u><i>Basse</i></u> | <i>Fermé</i> | |
| Date et heure | <i>03/10/23 11h03</i> | | | | |

Prendre en charge un incident



Agent / Consulter les incidents pris en charge :

(#05) Aymen

GILLA FE - Cas d'utilisation

Utilisateur : Consulter les incidents pris en charge

Description

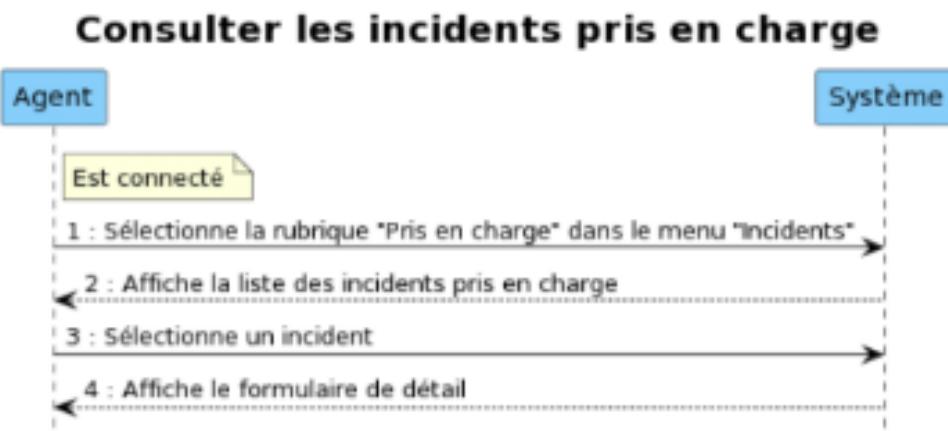
L'agent déjà connecté sélectionne la rubrique « Pris en charge » dans le menu « Incidents » et une liste d'incidents s'affiche alors. L'agent peut alors voir les détails de chaque incident sous forme de formulaire en cliquant sur celui-ci.

Maquette IHM

| GILLA | | |
|-----------------|------|-------------------|
| Type d'incident | Lieu | Etat d'avancement |
| Ecran cassé | A001 | pris en charge |
| Clavier cassé | F007 | pris en charge |
| | | |
| | | |

| GILLA | | |
|-------------------|----------------|--|
| Type d'incident | Ecran cassé | |
| Lieu | A001 | |
| Etat d'avancement | pris en charge | |
| Priorité | moyenne | |
| Clôturer | Oui/Non | |

Diagramme de séquence UML



Responsable / Gérer les agents :

(#06) Ilan

Responsable / Gérer les affectations :*(#07) Mathis***Description**

Le responsable déjà inscrit et connecter,

Il sélectionne la rubrique gérer les affectation suite à cela il clique sur un ticket puis affecte un agent au bon types d'incidents (types d'éléments)

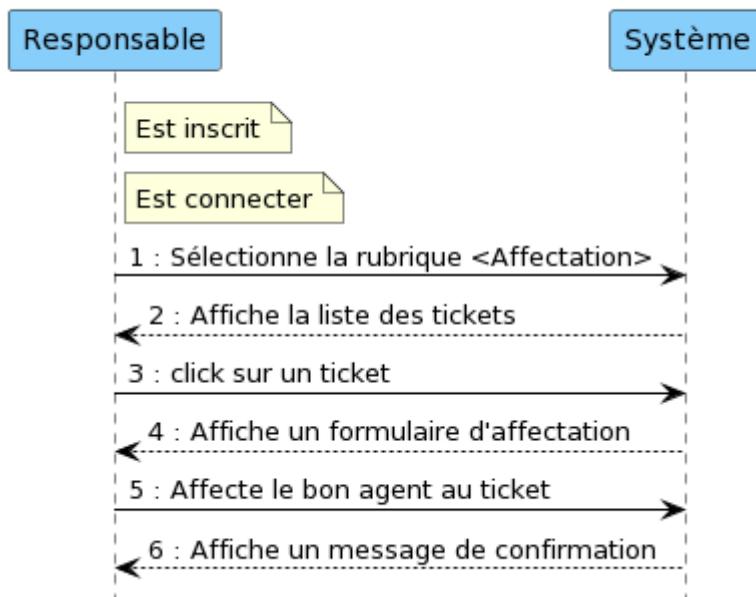
Maquette IHM

| GILLA | | | | |
|--------------|--------------------|-------------------|--------|------------|
| N° du ticket | Description | Types d'incidents | Statue | Date |
| 15 | PC en panne | Panne matériel | ouvert | 01/10/2023 |
| 16 | Bouteille renversé | Nettoyage | clos | 03/10/2023 |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |

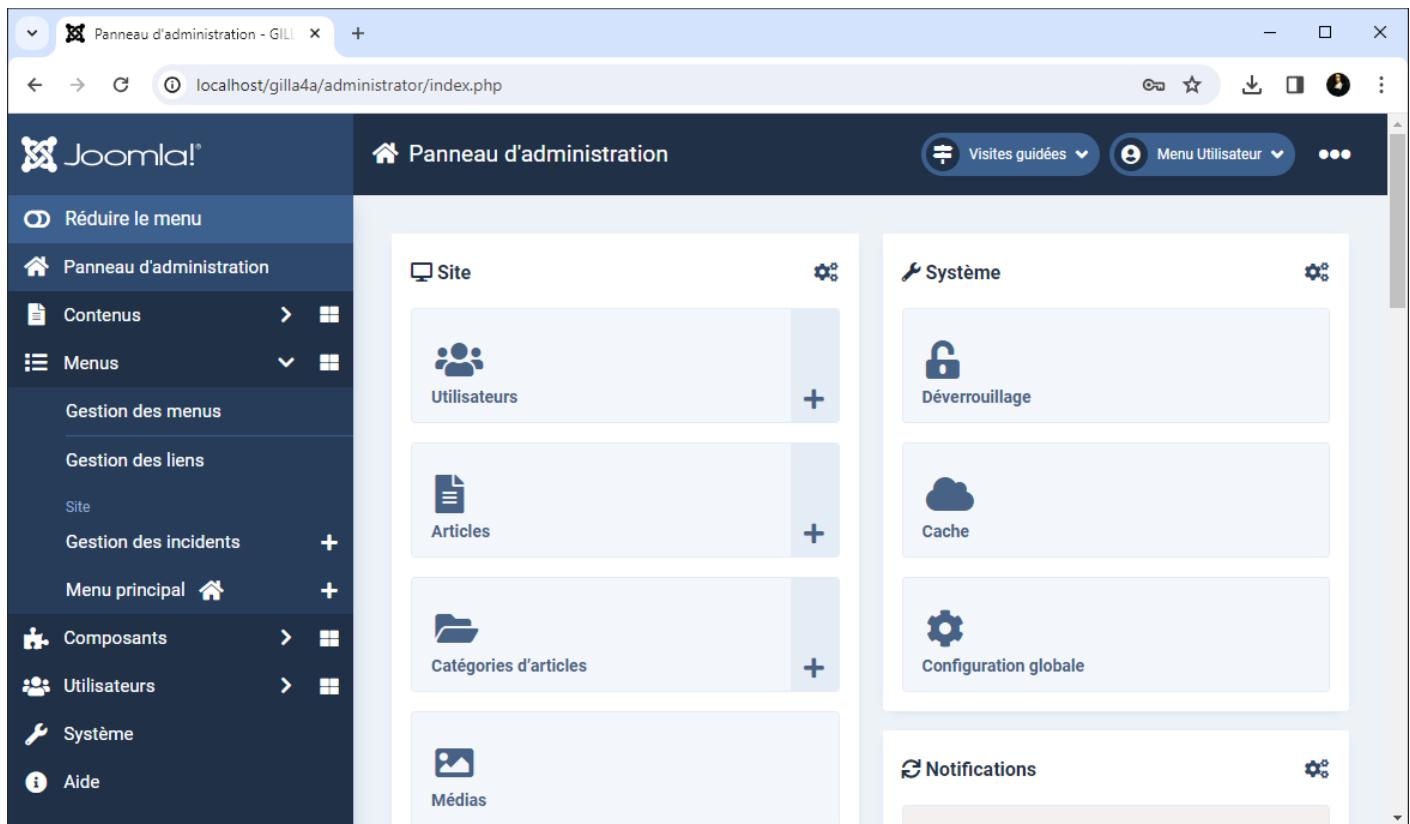
| N° du ticket | Attribution |
|--------------|-------------|
| 15 | Technicien |
| 16 | Ménage |
| | |

Diagramme de séquence UML

Gérer les affectations



Voici la page d'accueil du backend de Joomla 4 :
(#08) Oumar Vincent



The screenshot shows the Joomla 4 administrator dashboard. The left sidebar contains a navigation menu with items like 'Réduire le menu', 'Panneau d'administration', 'Contenus', 'Menus', 'Gestion des menus', 'Gestion des liens', 'Site', 'Gestion des incidents', 'Menu principal', 'Composants', 'Utilisateurs', 'Système', and 'Aide'. The main content area is divided into two columns. The left column, titled 'Site', contains icons for 'Utilisateurs', 'Articles', 'Catégories d'articles', and 'Médias', each with a '+' button to add new items. The right column, titled 'Système', contains icons for 'Déverrouillage', 'Cache', and 'Configuration globale', also with '+' buttons. Top navigation includes 'Visites guidées', 'Menu Utilisateur', and a three-dot menu. The browser address bar shows 'localhost/gilla4a/administrator/index.php'.

Après sélection du composant GILLA et de la rubrique « Prises en charge », on souhaite obtenir l'interface suivante :

(#08) Bayrone

The screenshot shows the Joomla! administrator dashboard with the GILLA component selected. The left sidebar shows various menu items like Contenus, Menus, Composants, and GILLA. The main content area displays a list of incidents with columns for Status, Incident ID, Agent, Start Date and Time, Priority, State, and ID. Three incidents are listed, each with a checked status icon and a blue link for the incident ID.

| | Status | Incident | Agent | Date et heure de début | Priorité | Etat | Id |
|--------------------------|-------------------------------------|-------------------------------|---------------|------------------------|-------------|--------------------|----|
| <input type="checkbox"/> | <input checked="" type="checkbox"/> | 20211123-0001 | Théo BELMONDO | 23-11-2021 08:20:00 | 2 - Moyenne | 2 - Pris en charge | 1 |
| <input type="checkbox"/> | <input checked="" type="checkbox"/> | 20211123-0002 | Marc DUPLAN | 23-11-2021 08:36:00 | 2 - Moyenne | 2 - Pris en charge | 2 |
| <input type="checkbox"/> | <input checked="" type="checkbox"/> | 20211123-0003 | Henry BOGART | 23-11-2021 08:39:00 | 1 - Haute | 2 - Pris en charge | 3 |

Interfaces utilisateurs du frontend GILLA

La page d'accueil du frontend GILLA devrait se présenter comme suit :

(#09) Gustave

The screenshot shows a web browser window with the title "#09) Gustave". The address bar displays "localhost/gilla4a/". The main content area shows the GILLA application's home page. The header features the GILLA logo and the text "Gestion des Incidents au Lycée Louis-Armand". A sidebar on the left contains links: Accueil, Lycée, Contact, Mon profil, and Evènements. The main content area has a large heading "Accueil". Below it, text states: "Toute personne travaillant au lycée (élève, étudiant, enseignant, AED, CPE, personnel administratif, agent technique) peut signaler un incident au service de maintenance du lycée." It then lists items to report: Bâtiment, Mobilier, and Informatique. Further down, it says "L'accès à ce service est réservé aux personnes disposant d'un compte sur ce site." and "La création d'un compte se fait avec une adresse email du lycée." At the bottom, there are two sections: "Gestion des incidents" (with links to Utilisateur, Incidents ouverts, Signaler un incident, and Mes incidents) and "Connexion" (with a welcome message "Bonjour, Aline DION" and a "Déconnexion" button).

Après connexion d'un agent, le menu « Gestion des incidents » et la page correspondante à la rubrique « Signaler un incident » devraient apparaître comme suit :

(#09) Davy

The screenshot shows a web application interface for managing incidents at Lycée Louis-Armand. The top navigation bar includes links for Accueil, Lycée, Contact, Mon profil, and Evènements. The main content area is titled 'Incident' and contains fields for Numéro, Date et heure d'ouverture, Emplacement, Type, Description, Date et heure de fermeture, and Commentaire de l'agent. To the right, a sidebar titled 'Gestion des incidents' provides links for Utilisateur (Incidents ouverts, Signaler un incident, Mes incidents), Agent (Incidents à gérer, Prises en charge), and Connexion (Déconnexion).

GILLA
Gestion des Incidents au Lycée Louis-Armand

Accueil Lycée Contact Mon profil Evènements

Incident

Numéro *

Date et heure d'ouverture *

Emplacement *

Type *

Description *

Date et heure de fermeture

Commentaire de l'agent

Gestion des incidents

Utilisateur :
Incidents ouverts
[Signaler un incident](#)
Mes incidents

Agent :
Incidents à gérer
Prises en charge

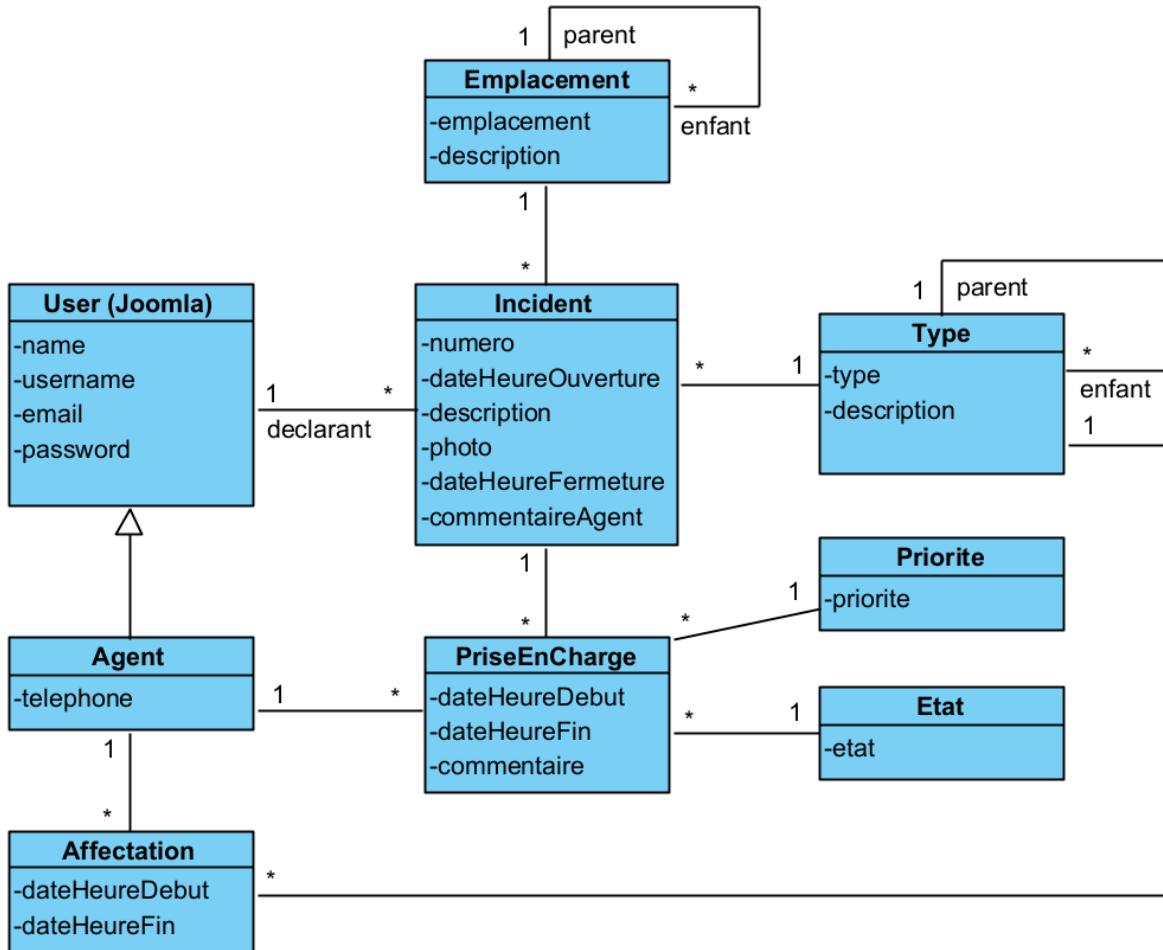
Connexion

Bonjour, Marc DUPLAN

Déconnexion

Modèle des données GILLA

Le modèle des données GILLA est décrit par le diagramme de classes UML suivant :



Les valeurs prédéfinies des attributs sont les suivantes :

| type | emplacement | priorite | etat |
|--------------------|-------------|---------------------------------------|---|
| Bâtiment | Bât A | 1 - Haute 2 - Moyenne 3 - Basse | 1 - Ouvert 2 - Pris en charge 3 - Fermé |
| - Aération | - Et A0 | | |
| - Eclairage | - A001 | | |
| - Chauffage | - ... | | |
| - Fenêtre | - A009 | | |
| - Mur | - Et A1 | | |
| - Plafond | - A101 | | |
| - Porte | - ... | | |
| - Prise de courant | - A117 | | |
| - Prise réseau | - Et A2 | | |
| - Sol | - A201 | | |
| Informatique | - ... | | |
| - Clavier | - A223 | | |
| - Ecran | - Et A3 | | |
| - PC portable | - A301 | | |
| - Souris | - ... | | |
| - Unité centrale | - A318 | | |
| - Vidéoprojecteur | - Et A4 | | |
| Mobilier | Bât B | | |
| - Armoire | Bât C | | |
| - Chaise | Bât D | | |
| - Table | Bât E | | |
| | Bât F | | |

Spécifications techniques

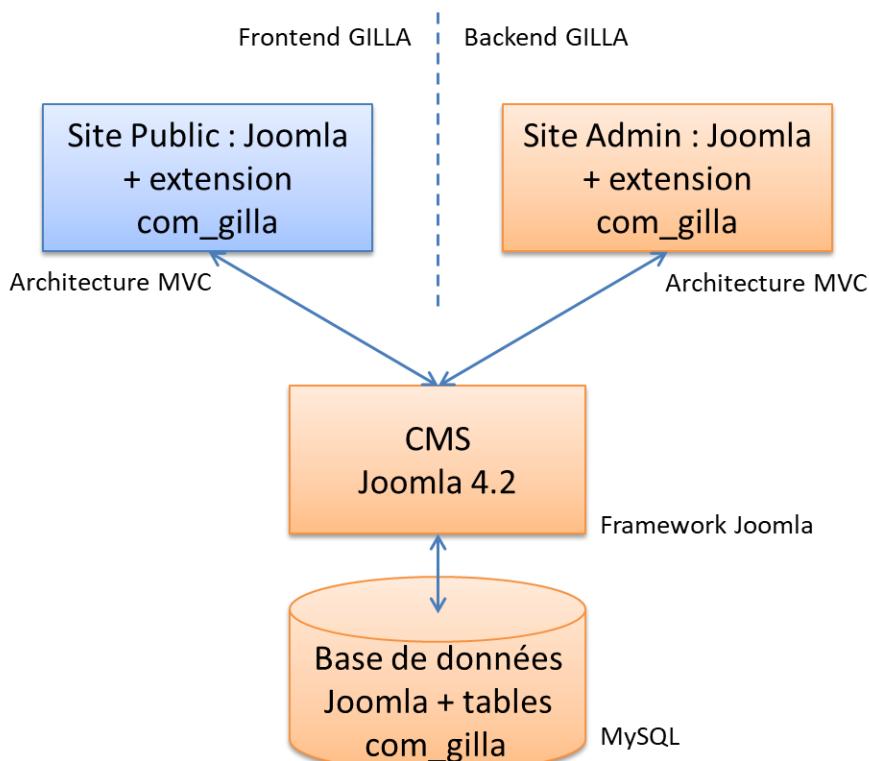
Framework et CMS Joomla

Afin de réduire les risques liés à la maîtrise d'œuvre d'un tel projet, l'équipe projet GILLA a retenu le principe d'une architecture logicielle ouverte (architecture MVC : Modèle-Vue-Contrôleur) avec Framework/CMS Joomla 4.4 et un développement PHP en deux phases :

1. Développement du backend GILLA et de la base de données « GILLA » sous la forme d'un composant « com_gilla » installable dans le CMS Joomla 4.4.
2. Développement du frontend GILLA avec le composant « com_gilla » pour implémenter les fonctionnalités attendues des utilisateurs.

La technologie Joomla a été choisie parmi les trois CMS (Content Management System : outil de gestion de contenu sur Internet) open-source et gratuits les plus populaires du marché : WordPress, Drupal et Joomla. Sa simplicité d'utilisation, la qualité de ses extensions et le dynamisme de ses communautés d'utilisateurs et de développeurs en France, en Europe et dans le monde assurent à Joomla un suivi et une évolution de très grande qualité.

Le schéma d'architecture logicielle est le suivant :



Les éléments coloriés en orange correspondent aux développements relatifs à la première phase et comprend les deux éléments du socle commun, la base données Joomla sous MySQL et l'installation de l'API Joomla 4.2, permettant ensuite l'intégration du composant « com_gilla » spécialement développé pour les besoins du site GILLA. L'élément colorié en bleu correspond à la deuxième phase du développement explicitée ci-dessus.

Environnements de développement et de tests

L'environnement de développement et de test qui a été mis en place est le suivant :

- Installation sur chaque PC étudiant de :
 - **XAMPP 8.1.10** (PHP 8.1.10, Apache 2.4.54, MariaDB 10.4.25, phpMyAdmin 5.2.0)
 - **Visual Studio Code 1.75.1**
 - **Joomla 4.4.2**
- Installation sur le serveur du labo SLAM :
 - Machine virtuelle Debian 8.2
 - Services Apache2, PHP et MySQL
 - Logiciel phpMyAdmin

L'environnement de production choisi a été une plateforme mutualisée de l'hébergeur OVH.com (offre Pro sur serveur LAMP) dont les détails sont donnés dans le chapitre déploiement.

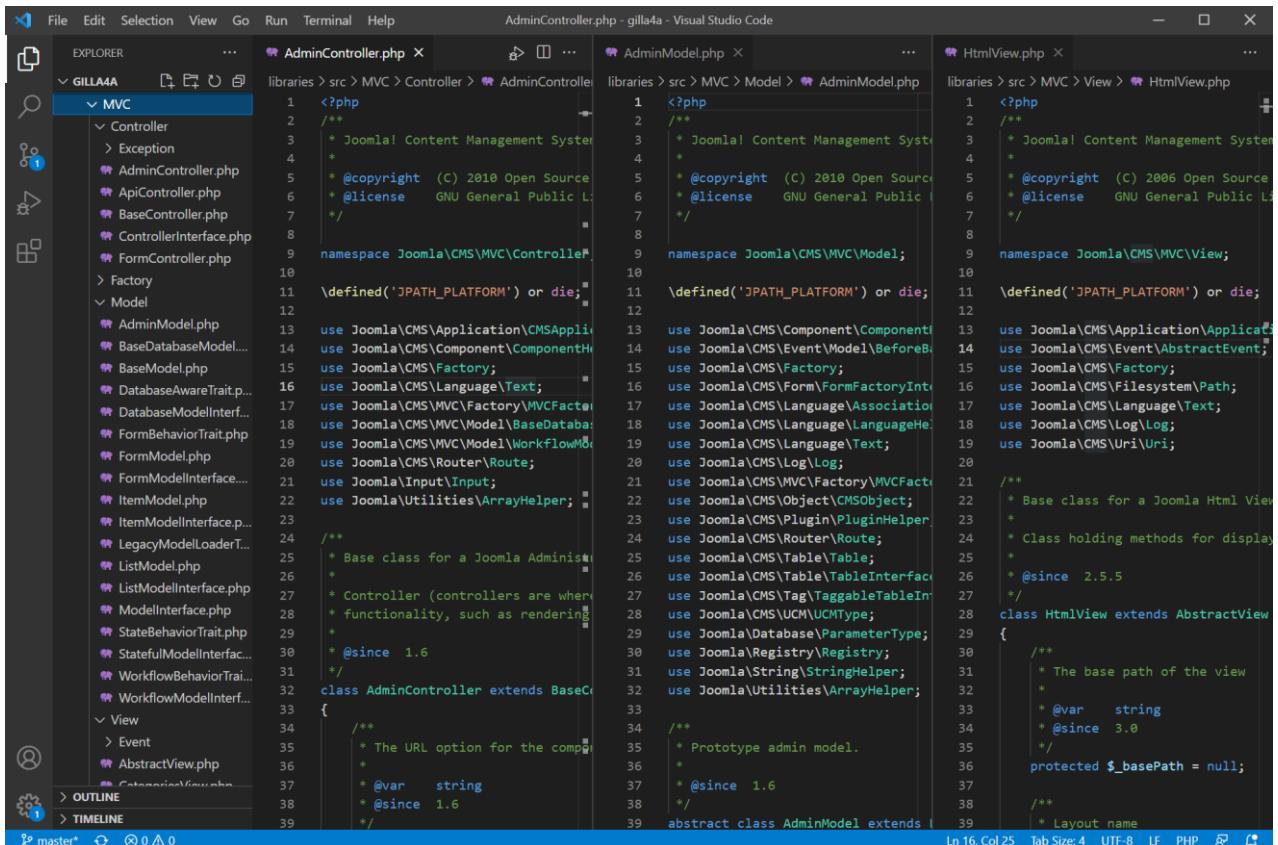
Conception de la solution

Architecture logicielle MVC

Le choix d'une architecture logicielle MVC répond aux besoins d'ouverture, de maintenance et d'évolutivité du site par une organisation standardisée du code source. Le but d'une telle architecture est de structurer le code, pour chaque cas d'utilisation ou bloc fonctionnel, en trois parties :

- Le **modèle** gère les données du site. Il récupère les informations dans la base de données, les organise et les assemble pour qu'elles puissent ensuite être traitées par le contrôleur et la vue. Cette partie contient donc les requêtes SQL organisées sous forme de fonctions PHP.
- La **vue** gère l'affichage. Elle organise la façon dont les données sont affichées à l'écran. On y trouve essentiellement du code HTML mais aussi quelques boucles et conditions PHP très simples.
- Le **contrôleur** gère la logique du code et les demandes utilisateurs. Le contrôleur récupère les données du modèle, les analyse/les traite, et renvoie les données à afficher à la vue. Le contrôleur contient exclusivement du code PHP, organisé sous forme de fonctions.

L'architecture du composant com_gilla respecte l'arborescence et les règles de nommage définies pour CMS Joomla 4.2. A chaque vue du backend ou du frontend correspond une structure MVC de trois classes qui héritent des classes MVC de base du CMS. Ainsi pour le backend, les classes AdminController, AdminModel et HtmlView suivantes seront étendues :



```

File Edit Selection View Go Run Terminal Help AdminController.php - gilla4a - Visual Studio Code
EXPLORER ... AdminController.php ... AdminModel.php ... HtmlView.php ...
GILLA4A ...
MVC ...
Controller ...
Exception ...
AdminController.php ...
ApiController.php ...
BaseController.php ...
ControllerInterface.php ...
FormController.php ...
Factory ...
Model ...
AdminController.php ...
BaseDatabaseModel.php ...
BaseModel.php ...
DatabaseAwareTrait.php ...
DatabaseModelInterface.php ...
FormBehaviorTrait.php ...
FormModel.php ...
FormModelInterface.php ...
ItemModel.php ...
ItemModelInterface.php ...
LegacyModelLoaderTrait.php ...
ListModel.php ...
ListModelInterface.php ...
ModelInterface.php ...
StateBehaviorTrait.php ...
StatefulModelInterface.php ...
WorkflowBehaviorTrait.php ...
WorkflowModelInterface.php ...
View ...
Event ...
AbstractView.php ...
CategoryView.php ...
OUTLINE ...
TIMELINE ...

```

```

AdminController.php
1 <?php
2 /**
3 * Joomla! Content Management System
4 *
5 * @copyright (C) 2010 Open Source
6 * @license GNU General Public L
7 */
8
9 namespace Joomla\CMS\MVC\Controller;
10
11 \defined('JPATH_PLATFORM') or die;
12
13 use Joomla\CMS\Application\CMSApplication;
14 use Joomla\CMS\Component\ComponentHelper;
15 use Joomla\CMS\Factory;
16 use Joomla\CMS\Language\Text;
17 use Joomla\CMS\MVC\Factory\MVCFactory;
18 use Joomla\CMS\MVC\Model\BaseDatabaseModel;
19 use Joomla\CMS\MVC\Model\WorkflowModel;
20 use Joomla\CMS\Router\Route;
21 use Joomla\Input\Input;
22 use Joomla\Utilities\ArrayHelper;
23
24 /**
25 * Base class for a Joomla Adminis
26 *
27 * Controller (controllers are where
28 * functionality, such as rendering
29 *
30 * @since 1.6
31 */
32 class AdminController extends BaseController
33 {
34 /**
35 * The URL option for the compo
36 *
37 * @var string
38 * @since 1.6
39 */

```

```

AdminModel.php
1 <?php
2 /**
3 * Joomla! Content Management System
4 *
5 * @copyright (C) 2010 Open Source
6 * @license GNU General Public L
7 */
8
9 namespace Joomla\CMS\MVC\Model;
10
11 \defined('JPATH_PLATFORM') or die;
12
13 use Joomla\CMS\Component\ComponentHelper;
14 use Joomla\CMS\Event\Model\BeforeEvent;
15 use Joomla\CMS\Factory;
16 use Joomla\CMS\Form\FormFactoryInt;
17 use Joomla\CMS\Language\Associatio
18 use Joomla\CMS\Language\LanguageHe
19 use Joomla\CMS\Language\Text;
20 use Joomla\CMS\Log\Log;
21
22 use Joomla\CMS\MVC\Factory\MVCFact
23 use Joomla\CMS\Object\CMSObject;
24 use Joomla\CMS\Plugin\PluginHelper;
25 use Joomla\CMS\Router\Route;
26 use Joomla\CMS\Table\TableInterface;
27 use Joomla\CMS\Table\TaggableTableIn
28 use Joomla\CMS\UCM\UCMType;
29 use Joomla\Database\ParameterType;
30 use Joomla\Registry\Registry;
31 use Joomla\String\StringHelper;
32 use Joomla\Utilities\ArrayHelper;
33
34 /**
35 * Prototype admin model.
36 *
37 * @since 1.6
38 */
39 abstract class AdminModel extends

```

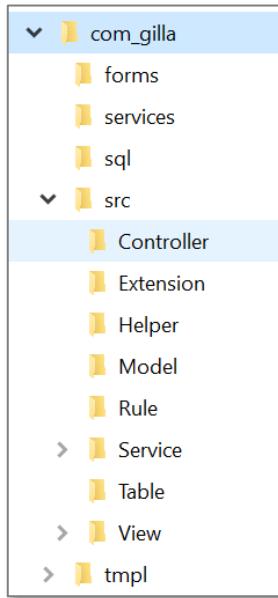
```

HtmlView.php
1 <?php
2 /**
3 * Joomla! Content Management System
4 *
5 * @copyright (C) 2006 Open Source
6 * @license GNU General Public L
7 */
8
9 namespace Joomla\CMS\MVC\View;
10
11 \defined('JPATH_PLATFORM') or die;
12
13 use Joomla\CMS\Application\Application;
14 use Joomla\CMS\Event\AbstractEvent;
15 use Joomla\CMS\Factory;
16 use Joomla\CMS\Filesystem\Path;
17 use Joomla\CMS\Language\Text;
18 use Joomla\CMS\Log\Log;
19 use Joomla\CMS\Uri\Uri;
20
21 /**
22 * Base class for a Joomla Html Vie
23 *
24 * Class holding methods for display
25 *
26 * @since 2.5.5
27 */
28 class HtmlView extends AbstractView
{
29
30 /**
31 * The base path of the view
32 *
33 * @var string
34 * @since 3.0
35 */
36 protected $_basePath = null;
37
38 /**
39 * Layout name

```

Architecture du backend de com_gilla

L'arborescence des dossiers du backend du composant com_gilla est la suivante :
(dossier **gilla/administrator/components/com_gilla**)



Les règles de nommage des fichiers php des contrôleurs, modèles et vues sont les suivantes :

- nom au pluriel pour les listes (ex : incidents)
- nom au singulier pour les formulaires de détail (ex : incident)

Les dossiers **Controller**, **Model**, **Table**, **View** et **tmpl** contiennent les fichiers suivants :

(#10) Radu

| Disque local (C:) > xampp > htdocs > gilla4a > administrator > components > com_gilla > src > Controller | | | | |
|--|--------------------------------|------------------|-------------|--------|
| | Nom | Modifié le | Type | Taille |
| forms | AffectationController.php | 26/03/2024 09:29 | Fichier PHP | 1 Ko |
| services | AffectationsController.php | 26/03/2024 09:29 | Fichier PHP | 1 Ko |
| sql | AgentController.php | 26/03/2024 09:29 | Fichier PHP | 1 Ko |
| src | AgentsController.php | 26/03/2024 09:29 | Fichier PHP | 1 Ko |
| Controller | DisplayController.php | 26/03/2024 09:29 | Fichier PHP | 1 Ko |
| Extension | EmplacementController.php | 26/03/2024 09:29 | Fichier PHP | 1 Ko |
| Helper | EmplacementsController.php | 26/03/2024 09:29 | Fichier PHP | 1 Ko |
| Model | EtatsController.php | 26/03/2024 09:29 | Fichier PHP | 1 Ko |
| Rule | Event_userController.php | 26/03/2024 09:29 | Fichier PHP | 1 Ko |
| Service | Event_usersController.php | 26/03/2024 09:29 | Fichier PHP | 3 Ko |
| Table | EventController.php | 26/03/2024 09:29 | Fichier PHP | 1 Ko |
| View | EventsController.php | 26/03/2024 09:29 | Fichier PHP | 1 Ko |
| tmpl | IncidentController.php | 26/03/2024 09:29 | Fichier PHP | 1 Ko |
| com_guidedtours | IncidentsController.php | 26/03/2024 09:29 | Fichier PHP | 1 Ko |
| com_installer | Prise_en_chargeController.php | 26/03/2024 09:29 | Fichier PHP | 1 Ko |
| com_joomlaupdate | Prise_En_ChargesController.php | 26/03/2024 09:29 | Fichier PHP | 1 Ko |
| com_languages | TypeController.php | 26/03/2024 09:29 | Fichier PHP | 1 Ko |
| | TypesController.php | 26/03/2024 09:29 | Fichier PHP | 1 Ko |

| Ce PC > Disque local (C:) > xampp > htdocs > gilla4a > administrator > components > com_gilla > src > Model | | | | |
|---|---------------------------|------------------|-------------|--------|
| | Nom | Modifié le | Type | Taille |
| forms | AffectationModel.php | 26/03/2024 09:29 | Fichier PHP | 6 Ko |
| services | AffectationsModel.php | 26/03/2024 09:29 | Fichier PHP | 5 Ko |
| sql | AgentModel.php | 26/03/2024 09:29 | Fichier PHP | 6 Ko |
| src | AgentsModel.php | 26/03/2024 09:29 | Fichier PHP | 5 Ko |
| Controller | EmplacementModel.php | 26/03/2024 09:29 | Fichier PHP | 6 Ko |
| Extension | EmplacementsModel.php | 26/03/2024 09:29 | Fichier PHP | 5 Ko |
| Helper | EtatsModel.php | 26/03/2024 09:29 | Fichier PHP | 5 Ko |
| Model | Event_userModel.php | 26/03/2024 09:29 | Fichier PHP | 5 Ko |
| Rule | Event_usersModel.php | 26/03/2024 09:29 | Fichier PHP | 5 Ko |
| Service | EventModel.php | 26/03/2024 09:29 | Fichier PHP | 6 Ko |
| Table | EventsModel.php | 26/03/2024 09:29 | Fichier PHP | 5 Ko |
| View | IncidentModel.php | 26/03/2024 09:29 | Fichier PHP | 6 Ko |
| tmpl | IncidentsModel.php | 26/03/2024 09:29 | Fichier PHP | 5 Ko |
| com_guidedtours | Prise_en_chargeModel.php | 26/03/2024 09:29 | Fichier PHP | 6 Ko |
| com_installer | Prise_En_ChargesModel.php | 26/03/2024 09:29 | Fichier PHP | 5 Ko |
| com_joomlaupdate | TypeModel.php | 26/03/2024 09:29 | Fichier PHP | 6 Ko |
| | TypesModel.php | 26/03/2024 09:29 | Fichier PHP | 5 Ko |

| Ce PC > Disque local (C:) > xampp > htdocs > gilla4a > administrator > components > com_gilla > src > Table | | | | |
|---|--------------------------|------------------|-------------|--------|
| | Nom | Modifié le | Type | Taille |
| Service | AffectationsTable.php | 26/03/2024 09:29 | Fichier PHP | 3 Ko |
| Table | AgentsTable.php | 26/03/2024 09:29 | Fichier PHP | 3 Ko |
| View | EmplacementsTable.php | 26/03/2024 09:29 | Fichier PHP | 3 Ko |
| tmpl | EtatsTable.php | 26/03/2024 09:29 | Fichier PHP | 3 Ko |
| com_guidedtours | Event_usersTable.php | 26/03/2024 09:29 | Fichier PHP | 2 Ko |
| com_installer | EventsTable.php | 26/03/2024 09:29 | Fichier PHP | 3 Ko |
| com_joomlaupdate | IncidentsTable.php | 26/03/2024 09:29 | Fichier PHP | 3 Ko |
| com_languages | Prise_en_chargeTable.php | 26/03/2024 09:29 | Fichier PHP | 3 Ko |
| com_login | TypesTable.php | 26/03/2024 09:29 | Fichier PHP | 3 Ko |

Ce PC > Disque local (C:) > xampp > htdocs > gilla4a > administrator > components > com_gilla > src > View >

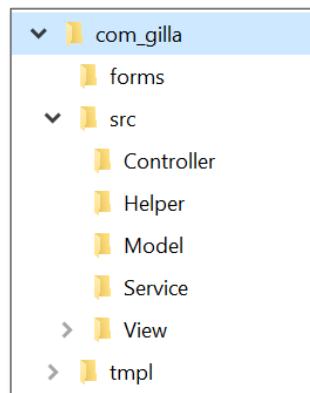
| Nom | Modifié le | Type | Taille |
|------------------|------------------|---------------------|--------|
| Affection | 26/03/2024 09:29 | Dossier de fichiers | |
| Affectations | 26/03/2024 09:29 | Dossier de fichiers | |
| Agent | 26/03/2024 09:29 | Dossier de fichiers | |
| Agents | 26/03/2024 09:29 | Dossier de fichiers | |
| Emplacement | 26/03/2024 09:29 | Dossier de fichiers | |
| Emplacements | 26/03/2024 09:29 | Dossier de fichiers | |
| Etats | 26/03/2024 09:29 | Dossier de fichiers | |
| Event | 26/03/2024 09:29 | Dossier de fichiers | |
| Event_user | 26/03/2024 09:29 | Dossier de fichiers | |
| Event_users | 26/03/2024 09:29 | Dossier de fichiers | |
| Events | 26/03/2024 09:29 | Dossier de fichiers | |
| Incident | 26/03/2024 09:29 | Dossier de fichiers | |
| Incidents | 26/03/2024 09:29 | Dossier de fichiers | |
| Pprise_en_charge | 26/03/2024 09:29 | Dossier de fichiers | |
| prise_en_charges | 26/03/2024 09:29 | Dossier de fichiers | |
| Type | 26/03/2024 09:29 | Dossier de fichiers | |
| Types | 26/03/2024 09:29 | Dossier de fichiers | |

Ce PC > Disque local (C:) > xampp > htdocs > gilla4a > administrator > components > com_gilla > tmpl >

| Nom | Modifié le | Type |
|------------------|------------------|---------------------|
| affection | 26/03/2024 09:29 | Dossier de fichiers |
| Affectations | 26/03/2024 09:29 | Dossier de fichiers |
| Agent | 26/03/2024 09:29 | Dossier de fichiers |
| agents | 26/03/2024 09:29 | Dossier de fichiers |
| Emplacement | 26/03/2024 09:29 | Dossier de fichiers |
| Emplacements | 26/03/2024 09:29 | Dossier de fichiers |
| etats | 26/03/2024 09:29 | Dossier de fichiers |
| event | 26/03/2024 09:29 | Dossier de fichiers |
| event_user | 26/03/2024 09:29 | Dossier de fichiers |
| event_users | 26/03/2024 09:29 | Dossier de fichiers |
| events | 26/03/2024 09:29 | Dossier de fichiers |
| incident | 26/03/2024 09:29 | Dossier de fichiers |
| Incidents | 26/03/2024 09:29 | Dossier de fichiers |
| Pprise_en_charge | 26/03/2024 09:29 | Dossier de fichiers |
| prise_en_Charges | 26/03/2024 09:29 | Dossier de fichiers |
| type | 26/03/2024 09:29 | Dossier de fichiers |
| types | 26/03/2024 09:29 | Dossier de fichiers |

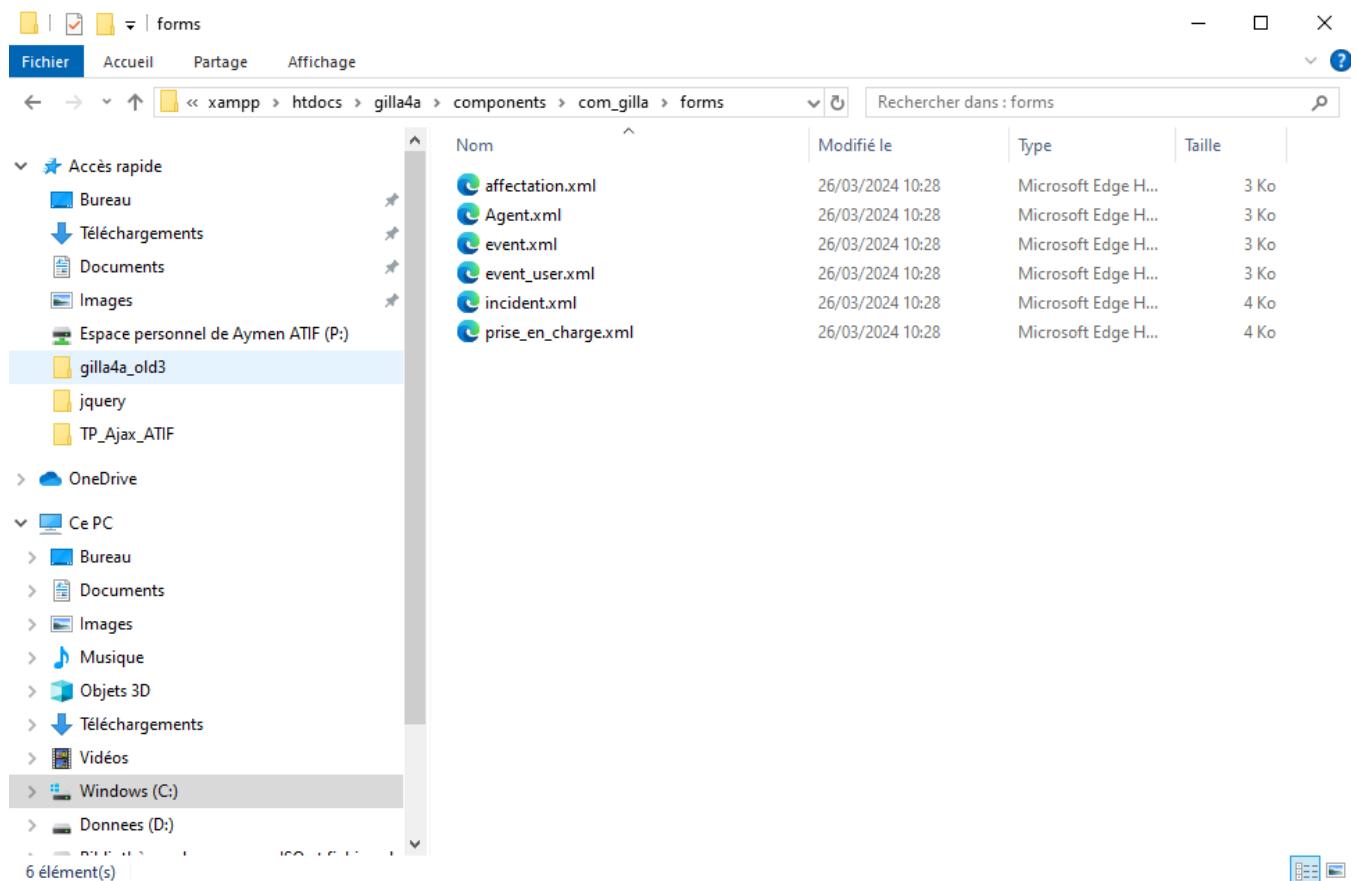
Architecture du frontend de com_gilla

L’arborescence du frontend du composant com_gilla est la suivante :
 (dossier **gilla/components/com_gilla**)



Les dossiers **forms**, **Controller**, **Model**, **View** et **tmpl** contiennent les fichiers suivants :

(#11) Aymen



Controller

| Nom | Modifié le | Type | Taille |
|-------------------------------|------------------|-------------|--------|
| AffectationController.php | 26/03/2024 10:28 | Fichier PHP | 5 Ko |
| AgentController.php | 26/03/2024 10:28 | Fichier PHP | 5 Ko |
| DisplayController.php | 26/03/2024 10:28 | Fichier PHP | 1 Ko |
| Event_userController.php | 26/03/2024 10:28 | Fichier PHP | 5 Ko |
| EventController.php | 26/03/2024 10:28 | Fichier PHP | 5 Ko |
| IncidentController.php | 26/03/2024 10:28 | Fichier PHP | 5 Ko |
| Prise_en_chargeController.php | 26/03/2024 10:28 | Fichier PHP | 5 Ko |

Model

| Nom | Modifié le | Type | Taille |
|----------------------------|------------------|-------------|--------|
| Affectation_eModel.php | 26/03/2024 10:28 | Fichier PHP | 5 Ko |
| AffectationModel.php | 26/03/2024 10:28 | Fichier PHP | 4 Ko |
| AffectationsModel.php | 26/03/2024 10:28 | Fichier PHP | 7 Ko |
| Agent_eModel.php | 26/03/2024 10:28 | Fichier PHP | 5 Ko |
| AgentModel.php | 26/03/2024 10:28 | Fichier PHP | 3 Ko |
| AgentsModel.php | 26/03/2024 10:28 | Fichier PHP | 6 Ko |
| Event_eModel.php | 26/03/2024 10:28 | Fichier PHP | 5 Ko |
| Event_user_eModel.php | 26/03/2024 10:28 | Fichier PHP | 5 Ko |
| EventModel.php | 26/03/2024 10:28 | Fichier PHP | 4 Ko |
| EventsModel.php | 26/03/2024 10:28 | Fichier PHP | 7 Ko |
| Incident_eModel.php | 26/03/2024 10:28 | Fichier PHP | 5 Ko |
| IncidentModel.php | 26/03/2024 10:28 | Fichier PHP | 3 Ko |
| IncidentsModel.php | 26/03/2024 10:28 | Fichier PHP | 8 Ko |
| Prise_en_charge_eModel.php | 26/03/2024 10:28 | Fichier PHP | 5 Ko |
| Prise_en_chargeModel.php | 26/03/2024 10:28 | Fichier PHP | 3 Ko |
| Prise_en_chargesModel.php | 26/03/2024 10:28 | Fichier PHP | 7 Ko |

Fichier Accueil Partage Affichage

« htdocs > gilla4a > components > com_gilla > src > View

Rechercher dans : View

Nom Modifié le Type Taille

| | | | |
| --- | --- | --- | --- |
| Affection | 26/03/2024 10:28 | Dossier de fichiers | |
| Affection_e | 26/03/2024 10:28 | Dossier de fichiers | |
| Affectations | 26/03/2024 10:28 | Dossier de fichiers | |
| Agent | 26/03/2024 10:28 | Dossier de fichiers | |
| Agent_e | 26/03/2024 10:28 | Dossier de fichiers | |
| Agents | 26/03/2024 10:28 | Dossier de fichiers | |
| Event | 26/03/2024 10:28 | Dossier de fichiers | |
| Event_e | 26/03/2024 10:28 | Dossier de fichiers | |
| Event_user_e | 26/03/2024 10:28 | Dossier de fichiers | |
| Events | 26/03/2024 10:28 | Dossier de fichiers | |
| Incident | 26/03/2024 10:28 | Dossier de fichiers | |
| Incident_e | 26/03/2024 10:28 | Dossier de fichiers | |
| Incidents | 26/03/2024 10:28 | Dossier de fichiers | |
| Prise_en_charge | 26/03/2024 10:28 | Dossier de fichiers | |
| Prise_en_charge_e | 26/03/2024 10:28 | Dossier de fichiers | |
| Prise_en_charges | 26/03/2024 10:28 | Dossier de fichiers | |

Fichier Accueil Partage Affichage

« xampp > htdocs > gilla4a > components > com_gilla > tmpl

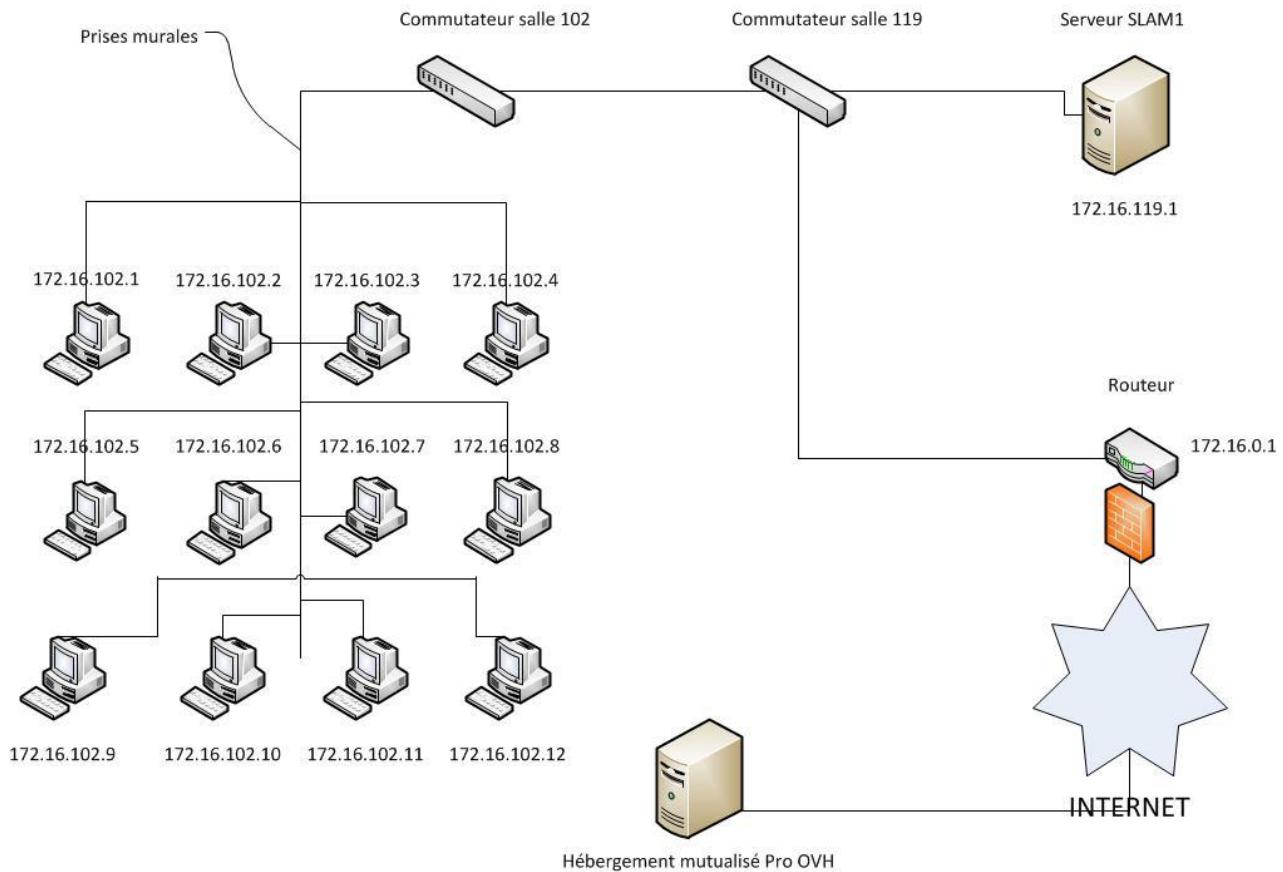
Rechercher dans : tmpl

Nom Modifié le Type Taille

| | | | |
| --- | --- | --- | --- |
| affectation | 26/03/2024 10:28 | Dossier de fichiers | |
| affectation_e | 26/03/2024 10:28 | Dossier de fichiers | |
| Affectations | 26/03/2024 10:28 | Dossier de fichiers | |
| Agent | 26/03/2024 10:28 | Dossier de fichiers | |
| Agent_e | 26/03/2024 10:28 | Dossier de fichiers | |
| agents | 26/03/2024 10:28 | Dossier de fichiers | |
| event | 26/03/2024 10:28 | Dossier de fichiers | |
| event_e | 26/03/2024 10:28 | Dossier de fichiers | |
| event_user_e | 26/03/2024 10:28 | Dossier de fichiers | |
| events | 26/03/2024 10:28 | Dossier de fichiers | |
| incident | 26/03/2024 10:28 | Dossier de fichiers | |
| incident_e | 26/03/2024 10:28 | Dossier de fichiers | |
| Incidents | 26/03/2024 10:28 | Dossier de fichiers | |
| prise_en_charge | 26/03/2024 10:28 | Dossier de fichiers | |
| prise_en_charge_e | 26/03/2024 10:28 | Dossier de fichiers | |
| prise_en_charges | 26/03/2024 10:28 | Dossier de fichiers | |

Architecture matérielle

L'infrastructure réseau mise en œuvre pour le développement de GILLA est la suivante :



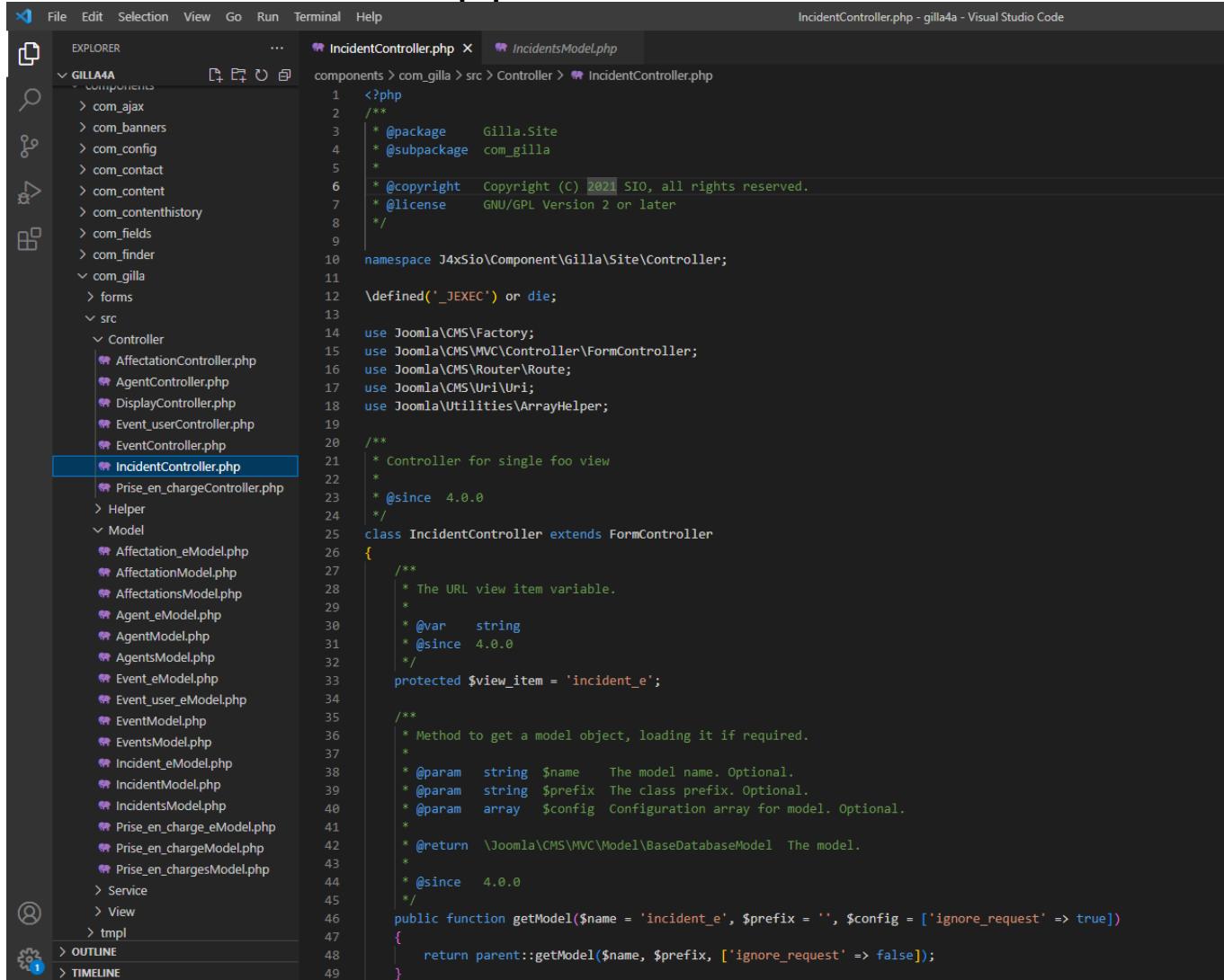
Codage du composant com_gilla

Codage MVC du backend de com_gilla

Les contrôleurs, modèles et vues constituant le code PHP du composant com_gilla selon l’arborescence et les règles de nommage des fichiers définis plus-haut sont conçus comme des classes PHP héritant des classes correspondantes du CMS Joomla.

Par exemple, la vue **liste des incidents** est constituée des fichiers de code suivants :
(#08) Malik

Contrôleur <> IncidentsController.php > :



The screenshot shows the Visual Studio Code interface with the file `IncidentsController.php` open. The left sidebar displays the project structure under the folder `GILLA4A`, which contains components like `com_ajax`, `com_banners`, `com_config`, etc., and a `src` folder containing `Controller`. Inside `Controller`, the file `IncidentsController.php` is selected. The main editor pane shows the PHP code for the controller:

```

<?php
/*
 * @package     Gilla.Site
 * @subpackage  com_gilla
 *
 * @copyright  Copyright (C) 2021 SIO, all rights reserved.
 * @license    GNU/GPL Version 2 or later
 */
namespace J4xSio\Component\Gilla\Site\Controller;

defined('_JEXEC') or die;

use Joomla\CMS\Factory;
use Joomla\CMS\MVC\Controller\FormController;
use Joomla\CMS\Router\Route;
use Joomla\CMS\Uri\Uri;
use Joomla\Utilities\ArrayHelper;

/**
 * Controller for single foo view
 *
 * @since 4.0.0
 */
class IncidentController extends FormController
{
    /**
     * The URL view item variable.
     *
     * @var string
     * @since 4.0.0
     */
    protected $view_item = 'incident_e';

    /**
     * Method to get a model object, loading it if required.
     *
     * @param string $name    The model name. Optional.
     * @param string $prefix  The class prefix. Optional.
     * @param array   $config Configuration array for model. Optional.
     *
     * @return \Joomla\CMS\MVC\Model\BaseDatabaseModel The model.
     * @since 4.0.0
     */
    public function getModel($name = 'incident_e', $prefix = '', $config = ['ignore_request' => true])
    {
        return parent::getModel($name, $prefix, ['ignore_request' => false]);
    }
}

```

Modèle <> IncidentsModel.php > :

```

File Edit Selection View Go Run Terminal Help
IncidentsModel.php - gilla4a - Visual Studio Code

EXPLORER IncidentController.php IncidentsModel.php
components > com_gilla > src > Model > IncidentsModel.php

1 <?php
2 /**
3 * @package Gilla.Site
4 * @subpackage com_gilla
5 *
6 * @copyright Copyright (C) 2021 SIO, all rights reserved.
7 * @license GNU/GPL Version 2 or later
8 */
9
10 namespace J4xSio\Component\Gilla\Site\Model;
11
12 defined('_JEXEC') or die;
13
14 use Joomla\CMS\Component\ComponentHelper;
15 use Joomla\CMS\Factory;
16 use Joomla\CMS\Language\Multilanguage;
17 use Joomla\CMS\WVC\Model\ListModel;
18 // use Joomla\Database\ParameterType;
19 // use Joomla\Registry\Registry;
20
21 /**
22 * Gilla component incidents model.
23 *
24 * @since 1.6
25 */
26 class IncidentsModel extends ListModel
27 {
28     /**
29      * Constructor.
30      *
31      * @param array $config An optional associative array of configuration settings.
32      *
33      * @see \JController
34      * @since 1.6
35      */
36     public function __construct($config = array())
37     {
38         if (empty($config['filter_fields']))
39         {
40             $config['filter_fields'] = array(
41                 'id', 'a.id',
42                 'numero', 'a.numero',
43                 'date_heure_ouverture', 'a.date_heure_ouverture',
44                 'description', 'a.description',
45                 'photo', 'a.photo',
46                 'date_heure_fermeture', 'a.date_heure_fermeture',
47                 'commentaire_agent', 'a.commentaire_agent',
48                 'users_id', 'a.users_id',
49                 'emplacements_id', 'a.emplacements_id',

```

Table « IncidentsTable.php » :

The screenshot shows a code editor interface with the following details:

- File Menu:** File, Edit, Selection, View, Go, Run, Terminal, Help.
- Explorer:** Shows the project structure under "GILLA4A". Key components include com_content, com_contenthistory, com_cpanel, com_fields, com_finder, com_gilla (which contains forms, services, sql), and src (Controller, Extension, Helper, Model, Rule, Service). The "Table" component is expanded, showing files like AffectionTable.php, AgentsTable.php, EmplacementsTable.php, EtatsTable.php, Event_usersTable.php, EventsTable.php, IncidentsTable.php (the current file being edited), Prise_en_chargesTable.php, and TypesTable.php.
- Terminal:** IncidentsTable.php
- Code Editor:** The file IncidentsTable.php is open. The code is a PHP class definition for "IncidentsTable" extending "Table". It includes annotations for package,subpackage, copyright, and license. The constructor __construct takes a DatabaseDriver object. A generateAlias() method is also present.

```

1 <?php
2 /**
3 * @package      Gilla\Administrator
4 * @subpackage   com_gilla
5 *
6 * @copyright   Copyright (C) 2021 SIO, all rights reserved.
7 * @license     GNU/GPL Version 2 or later
8 */
9
10 namespace J4xSio\Component\Gilla\Administrator\Table;
11
12 defined('JPATH_PLATFORM') or die;
13
14 use Joomla\CMS\Application\ApplicationHelper;
15 use Joomla\CMS\Table\Table;
16 use Joomla\CMS\Tag\TaggableTableInterface;
17 use Joomla\CMS\Tag\TaggableTableTrait;
18 use Joomla\Database\DatabaseDriver;
19 use Joomla\CMS\Language\Text;
20 use Joomla\Registry\Registry;
21 use Joomla\CMS\Factory;
22
23 /**
24 * Events table
25 *
26 * @since 1.5
27 */
28 class IncidentsTable extends Table
29 {
30     /**
31      * Constructor
32      *
33      * @param DatabaseDriver $db Database connector object
34      *
35      * @since 1.0
36      */
37     public function __construct(DatabaseDriver $db)
38     {
39         $this->typeAlias = 'com_gilla.incident';
40         parent::__construct('#_gilla_incidents', 'id', $db);
41     }
42
43     /**
44      * Generate a valid alias from title / date.
45      * Remains public to be able to check for duplicated alias before s
46      *
47      * @return string
48      */
49     public function generateAlias()

```

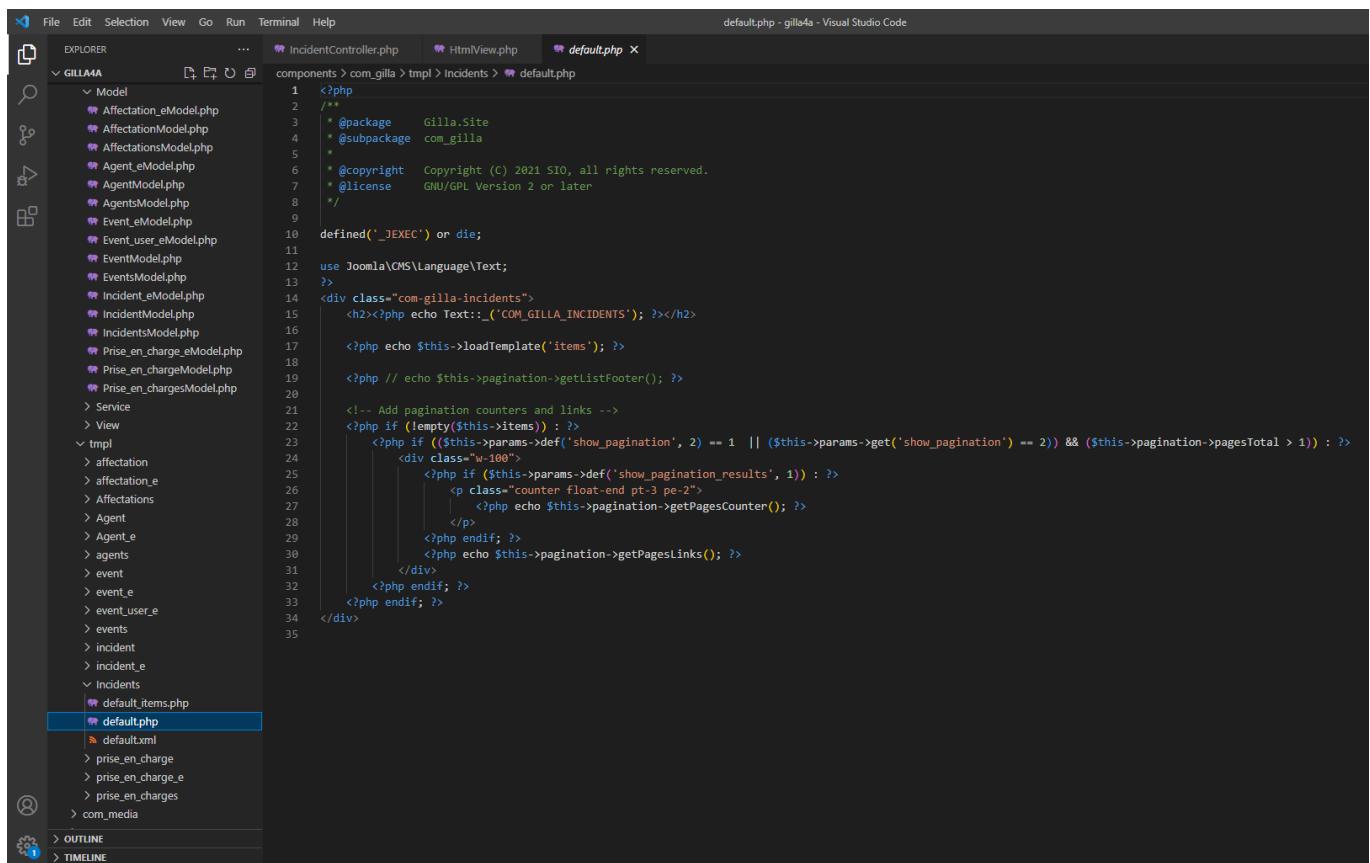
Vue « Incidents/HtmlView.php » :

```

1  ?>php
2  /**
3   * @package     Gillia.Site
4   * @subpackage  com_gilla
5   *
6   * @copyright   Copyright (C) 2021 SIO, all rights reserved.
7   * @license     GNU/GPL Version 2 or later
8   */
9
10  namespace J4xSio\Component\Gilla\Site\View\Incidents;
11
12  defined('_JEXEC') or die;
13
14  use Joomla\CMS\Factory;
15  //use Joomla\CMS\HTML\Helper;
16  //use Joomla\CMS\Language\Text;
17  use Joomla\CMS\MVC\View\GenericDataException;
18  use Joomla\CMS\MVC\View\HtmlView as BaseHtmlView;
19  use Joomla\CMS\Router\Route;
20  use J4xSio\Component\Gilla\Site\Helper\RouteHelper as GillaHelperRoute;
21
22 /**
23  * Incidents list view class
24  *
25  * @since 1.6
26  */
27 class HtmlView extends BaseHtmlView
28 {
29
30     /**
31      * The item model state
32      *
33      * @var \Joomla\Registry\Registry
34      * @since 1.6.0
35      */
36     protected $state;
37
38     /**
39      * The item details
40      *
41      * @var \ JObject
42      * @since 1.6.0
43      */
44     protected $items;
45
46     /**
47      * The pagination object
48      *
49      * @var \JPagination
49      * @since 1.6.0
</pre

```

Template < incidents/default.php > :



The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows the project structure under the 'GILLA' component, including Model, Service, View, tmpl, and tmpf sections.
- Editor:** Displays the 'default.php' file content, which is a PHP template for the 'com_gilla' component's 'incidents' view.
- Code:** The code includes Joomla CMS language text, a main heading, a loop for items, and pagination links. It uses Joomla's language system and includes logic for handling empty item lists and generating page links.

```

1 <?php
2 /**
3  * @package    Gilla.Site
4  * @subpackage com_gilla
5  *
6  * @copyright Copyright (C) 2021 SIO, all rights reserved.
7  * @license   GNU/GPL Version 2 or later
8  */
9
10 defined('_JEXEC') or die;
11
12 use Joomla\CMS\Language\Text;
13 >
14 <div class="com_gilla-incidents">
15 <h2><?php echo Text::_('COM_GILLA INCIDENTS'); ?></h2>
16
17 <?php echo $this->loadTemplate('items'); ?>
18
19 <?php // echo $this->pagination->getListFooter(); ?>
20
21 <!-- Add pagination counters and links -->
22 <?php if (!empty($this->items)) : ?>
23 <?php if ((isset($this->params->def('show_pagination', 2) == 1 || ($this->params->get('show_pagination') == 2)) && ($this->pagination->pagesTotal > 1)) : ?>
24 <div class="w-100">
25 <?php if ($this->params->def('show_pagination_results', 1)) : ?>
26 <div class="counter float-end pt-3 pe-2">
27 <p><?php echo $this->pagination->getPagesCounter(); ?>
28 </p>
29 <?php endif; ?>
30 <?php echo $this->pagination->getPagesLinks(); ?>
31 </div>
32 <?php endif; ?>
33 <?php endif; ?>
34 </div>
35

```

Codage MVC du frontend de com_gilla

Les contrôleurs, modèles et vues constituant le code PHP du composant com_gilla selon l’arborescence et les règles de nommage des fichiers définis plus-haut sont conçus comme des classes PHP héritant des classes correspondantes du CMS Joomla.

Par exemple, la vue d'**édition d'un incident** est constituée des fichiers de code suivants :
(#09) Ilan

Formulaire « incident.xml » :

Contrôleur « IncidentController.php » :

Modèle « Incident_eModel.php » :

Vue « Incident_e/HtmlView.php » :

Template « incident_e/edit.php » :

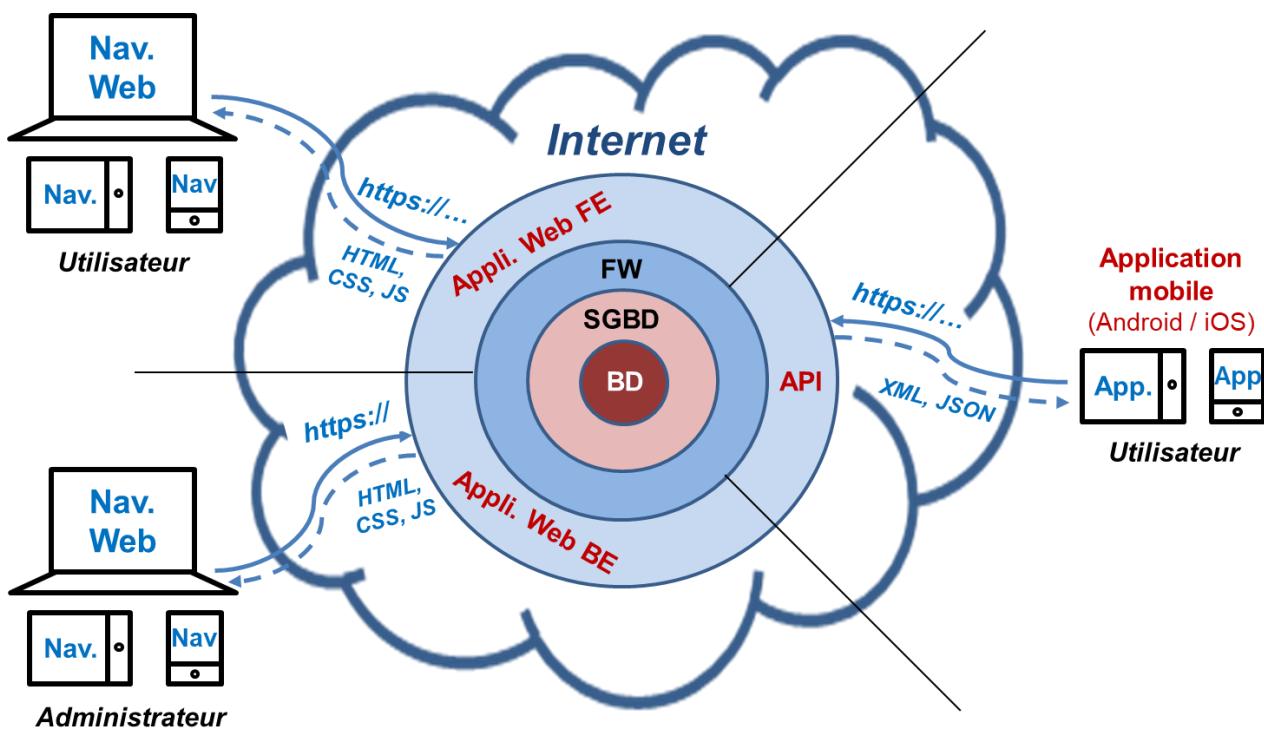
Application AndroidGilla

Architecture de la solution

Côté serveur, une API de test a été spécialement développée en PHP pour pouvoir consulter la base de données GILLA à travers le framework Joomla par le protocole http.

Côté client, l'application Android a été développée en Java pour permettre à l'utilisateur « client » disposant d'une tablette ou d'un smartphone Android de se connecter à l'API par le protocole http et consulter ensuite les offres et ses candidatures.

En environnement de développement, serveur et terminal Android sont connecté au même réseau local.



l'API du site GILLA dispose de cinq points d'entrée (*entry points*) de type CRUD (avec deux *Read* : liste et unitaire) par table (incidents ici) :

- POST gilla/incidents (+ body) : création d'un incident.
- GET gilla/incidents: consultation de la liste des incidents.
- GET gilla/incidents/{incident_id} : consultation de l'incident *incident_id*.
- PATCH gilla/incidents/{incident_id} (+ body) : modification de l'incident *incident_id*.
- DEL gilla/incidents/{incident_id} : suppression de l'incident *incident_id*.

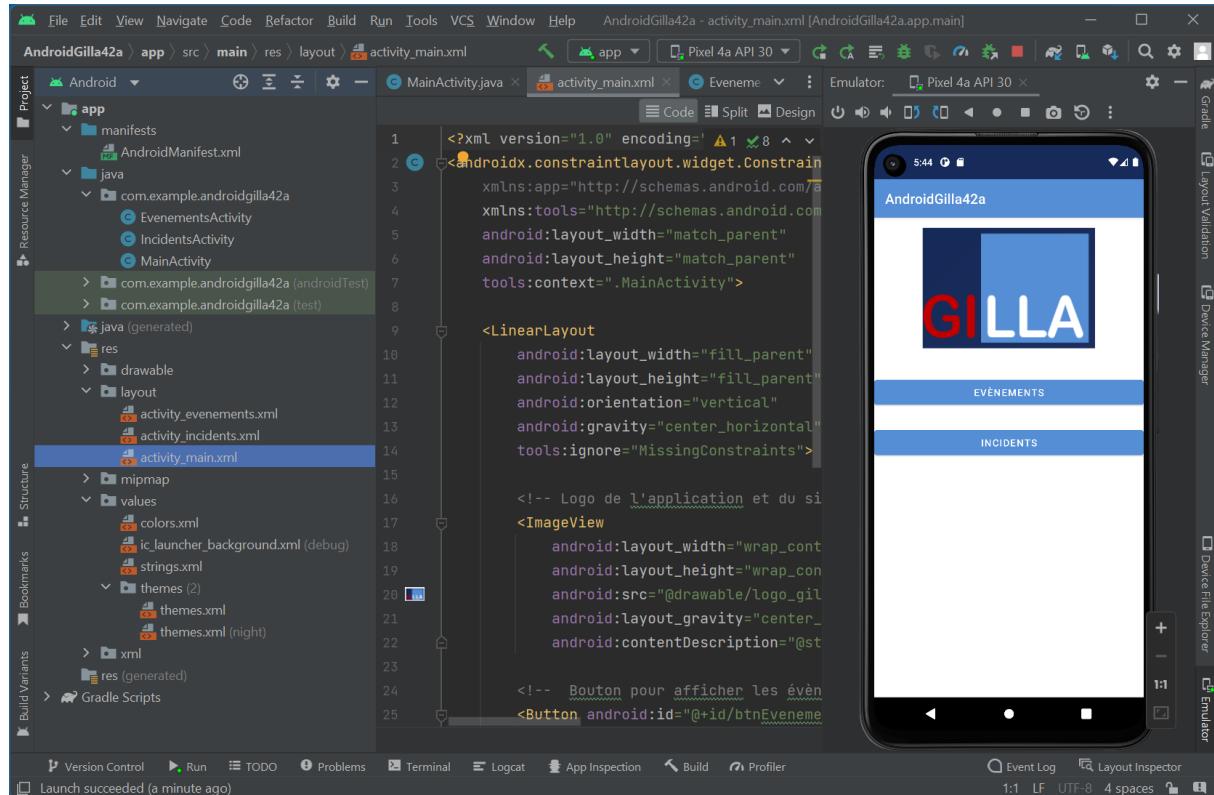
Android Studio

Conformément aux dernières recommandations de Google, l'environnement de développement Android Studio a été choisi pour concevoir et tester l'application AndroidGilla.

Android Studio est un IDE qui permet de développer des applications Android en Java, il bénéficie aussi de la technologie WYSIWYG (What You See Is What You Get) qui permet de voir et gérer la mise en page à côté de l'XML.

Structure et code de l'application

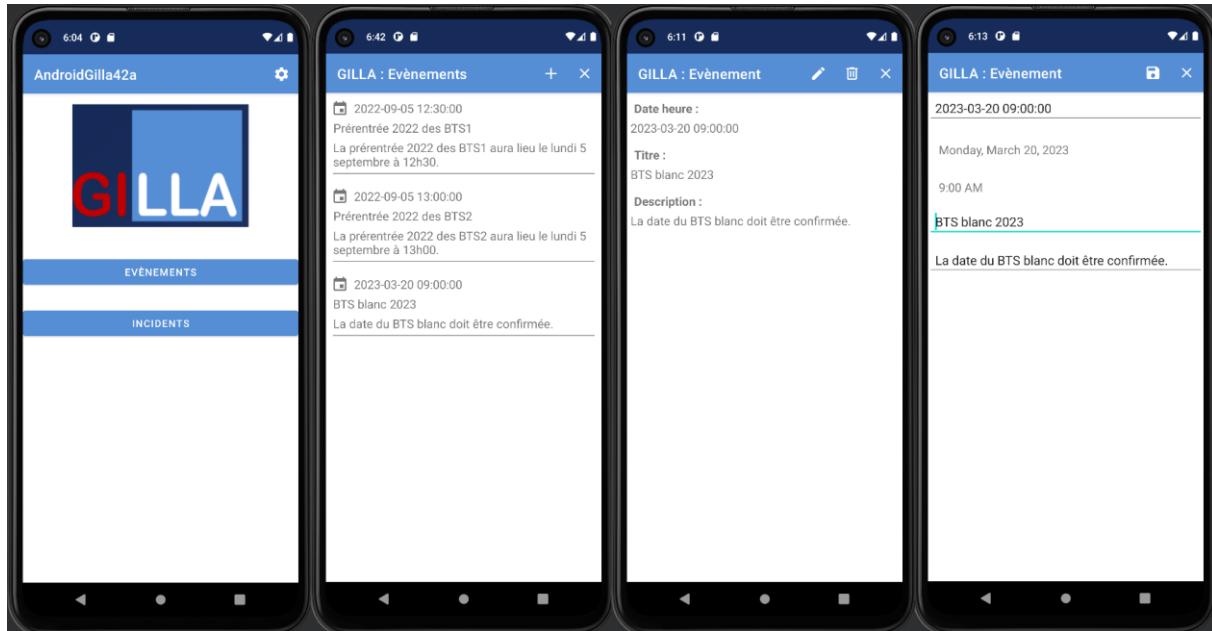
La structure de l'application AndroidGilla est la suivante :



L'arborescence « app » contient les dossiers « manifests », « java » qui contient les activités (contrôleurs et modèles codés en JAVA) et « res » qui contient notamment le dossier « layout » des écrans codés en XML (vues).

Tests de l'application

Une fois l'application lancée, la sélection dans le **menu principal** de « Evènements » affiche la **vue liste** et les **vues de détail** du 3^e évènement comme suit (de gauche à droite) :



Intégration et tests d'intégration du site **GILLA**

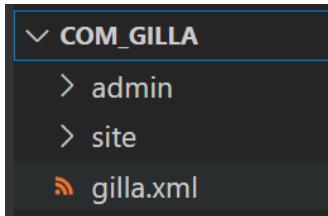
Création du composant packagé com_gilla.zip

Il s'agit de créer un composant packagé **com_gilla.zip** installable depuis le CMS Joomla 4.2. Le **manifeste gilla.xml** décrit les attributs et les dossiers et fichiers d'installation des tables et des arborescences du frontend et du backend du composant comme suit :

```

administrator > components > com_gilla > gilla.xml
1   <?xml version="1.0" encoding="UTF-8"?>
2   <extension type="component" method="upgrade">
3       <name>com_gilla</name>
4       <creationDate>Oct 2022</creationDate>
5       <author>SIO</author>
6       <copyright>Copyright (C) 2021-2022 SIO, all rights reserved.</copyright>
7       <license>GNU/GPL Version 2 or later - http://www.gnu.org/licenses/gpl-2.0.html</license>
8       <version>0.6.1</version>
9       <description>COM_GILLA_DESCRIPTION</description>
10      <namespace path="src">J4xSio\Component\Gilla</namespace>
11
12  >     <install>          <!-- Runs on install -->...
16  >     </install>
17  >     <uninstall>        <!-- Runs on uninstall -->...
21  >     </uninstall>
22
23     <!-- Frontend Site Section -->
24     <files folder="site">
25         <folder>forms</folder>
26         <folder>src</folder>
27         <folder>tmpl</folder>
28     </files>
29     <languages folder="site">
30         <language tag="en-GB">language/en-GB/com_gilla.ini</language>
31         <language tag="fr-FR">language/fr-FR/com_gilla.ini</language>
32     </languages>
33
34     <!-- Backend Site Section -->
35     <administration>
36         <menu img="class:default" link="option=com_gilla">com_gilla</menu>
37     >     <submenu> ...
47     >     </submenu>
48     >     <files folder="admin">...
56     >     </files>
57     >     <languages folder="admin">...
62     >     </languages>
63     </administration>
64
65     <!-- API Section -->
66     <api>
67         <files folder="api">
68             <folder>src</folder>
69         </files>
70     </api>
71 </extension>
```

Le dossier compressé contient ce fichier manifeste et les deux arborescence sources :



Test d'installation du composant packagé com_gilla.zip

Il s'agit de vérifier la bonne installation du composant **com_gilla.zip** à partir du menu du backend de Joomla : Système / Installation / Extensions. Un glisser-déposer du fichier com_gilla.zip dans le cadre centre ou une recherche du fichier sur le système déclenche l'installation du composant. La bonne installation est confirmée par un message sur fond vert : « Installation du composant réussie ».

The screenshot shows the Joomla administrator dashboard with the 'Extensions : Installer' page open. The left sidebar includes links for Contenus, Menus, Composants, Utilisateurs, Système, and Aide. The main content area has tabs for 'Gestion des extensions', 'Paramètres', and 'Aide'. The 'Archive à envoyer' tab is active. A central area features a dashed box for dragging files, a 'Ou rechercher le fichier' button, and a note about the 40.00 MB file size limit.

Test de l'API

Le test de l'API du composant com_gilla se fait avec le logiciel **Postman** et une collection de requêtes développée au format JSON puis importée dans ce logiciel.

Après paramétrage du Token obtenu dans le backend de GILLA (onglet « Jeton Joomla » du super utilisateur SLAM) et du chemin « base_path » définissant la racine du site, il est possible d'exécuter les requêtes prévues dans la collection :

The screenshot shows the Postman application interface. On the left, there's a sidebar with sections for Collections, APIs, Environments, Mock Servers, and Monitors. The main area is titled "Working locally in Scratch Pad. Switch to a Workspace". It displays a collection named "Joomla 4 Gilla Component Web Servic...". Under this collection, there are several requests listed:

- GET {{base_path}}/api/index.php/v1/...
- GET {{base_path}}/api/index.php/v1/...
- DEL {{base_path}}/api/index.php/v1/...
- POST {{base_path}}/api/index.php/v1/...
- PATCH {{base_path}}/api/index.php/v1/...

One request is currently selected: "GET {{base_path}}/api/index.php/v1/gilla/events". The "Headers" tab is active, showing two checked items: "Accept" with value "*/*" and "Accept-Encoding" with value "gzip, deflate, br". Below the headers, the "Body" section is set to "Pretty" view, showing a JSON response:

```

1
2   "links": {
3     "self": "http://192.168.1.69/gilla4a/api/index.php/v1/gilla/events"
4   },
5   "data": [
6     {
7       "type": "events",
8       "id": "1",
9       "attributes": {
10         "id": 1,
11         "title": "Prérentrée 2021 des BTS1",
12         "description": "La prérentrée 2021 des BTS1 aura lieu le vendredi 3 septembre à 14h00.",
13         "date_time": "2021-09-03 14:00:00"
14       }
15     }
]

```

The response status is 200 OK, with a duration of 1.441 ms and a size of 1.78 KB. There are buttons for "Save Response" and "Runner".

On peut ainsi analyser les réponses JSON envoyées par l'API du site GILLA.

Tests et recette fonctionnelle

Tests du backend GILLA

Le cahier de tests du backend GILLA comprend pour chaque cas d'utilisation identifié lors des spécifications une fiche de test selon modèle suivant :

(#01 : *BE com_gilla - Consulter Prises en charge*)

Tests du frontend GILLA

Les tests du frontend GILLA se sont déroulés selon le même principe que ceux du backend, à partir d'un cahier de tests composé de fiches de tests associées aux cas d'utilisation spécifiés dans le frontend, comme par exemple :

(#02 : *FE com_gilla – Consulter mes incidents*)

Tests de l'application AndroidGilla

Les tests de l'application AndroidGilla ont consisté à vérifier le bon affichage d'un certain nombre d'écrans comme par exemple :

(#03)

Déploiement

Le déploiement du site GILLA s'est fait sur une plateforme mutualisée LAMP hébergée chez OVH, après achat du nom de domaine disponible lla-sio.fr et de l'hébergement Pro, de la façon suivante :

1. Préparation des fichiers à télécharger :
 - Export de la base de données GILLA complète au format SQL (CMS Joomla + extension com_gilla).
 - Copie du fichier GILLA.sql à la racine de l'arborescence des fichiers du site.
2. Préparation de la plateforme d'hébergement :
 - Création du sous domaine GILLA dans www.llas-sio.fr.
 - Création du répertoire GILLA à la racine www du site (correspondant au sous-domaine).
 - Création de la base de données sous MySQL.
3. Téléchargement des fichiers :
 - Téléchargement de l'arborescence des fichiers par FTP dans le répertoire www du serveur OVH.
4. Installation de la base de données :
 - Import du fichier GILLA.sql (structure et contenu de la base de données) avec phpMyAdmin (serveur OVH).
5. Tests :
 - Test du site public : <https://gilla4a.sio.louis-armand.paris> (ou ...gilla4b...)
 - Test du site d'administration :
<https://gilla4a.sio.louis-armand.paris/administrator> (ou ...gilla4b...)