



PARDON

Oscar

SIO SLAM 2025

GSB NEWS

Projet Personnel Encadré

SOMMAIRE

1 / Fiche descriptive de réalisation professionnelle

2 / Origine du projet

A / Présentation de Galaxy Swiss Bourdin

B / Contexte de la demande

C / Contenu et fonctionnalités attendues

D / Planning prévisionnel et budget

3 / Préparation du projet

A / Schémas de l'application

B / Modèle conceptuel de données

C / Liste des tâches à réaliser

4 / Explication de la conception générale de l'application

A / Fonctionnalités principales

B / Rôles

5 / Explication de la conception technique de l'application

A / Technologies utilisées

B / Arborescence et architecture de l'application

C / Sécurité

D / Tests

6 / Bilan post-réalisation :

A / Planning réel de la réalisation de l'application

B / Versionning

C / Améliorations Futures de l'applications

7 / Documentation Utilisateur

Fiche descriptive de réalisation professionnelle

BTS SERVICES INFORMATIQUES AUX ORGANISATIONS		SESSION 2025
ANNEXE 9-1-B : Fiche descriptive de réalisation professionnelle (recto)		
Épreuve E6 - Conception et développement d'applications (option SLAM)		

DESCRIPTION D'UNE RÉALISATION PROFESSIONNELLE		N° réalisation : 1
Nom, prénom : PARDON, Oscar		N° candidat : 02148596825
Épreuve ponctuelle <input type="checkbox"/>	Contrôle en cours de formation <input checked="" type="checkbox"/>	Date : 28 / 03 / 2025
Organisation support de la réalisation professionnelle <p>La réalisation s'inscrit dans le cadre d'un Projet Personnalisé Encadré, fondé sur un contexte fourni par le centre de formation. Il concerne une entreprise pharmaceutique fictive, nommée Laboratoire Galaxy Swiss Bourdin (GSB).</p>		
Intitulé de la réalisation professionnelle <p>Application mobile de blog d'entreprise</p>		
Période de réalisation : mars 2025		Lieu : AFIP Formations
Modalité : Seul(e) <input checked="" type="checkbox"/>		En équipe <input type="checkbox"/>
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 réalisation² (ressources fournies, résultats attendus) <p>Ressources fournies : Contexte GSB de l'éducation nationale, normes de développement, description du contexte, ressources de formation.</p> <p>Résultats attendus : Réalisation d'une application mobile complète (Front end, Backend, API, Tests, Versioning, Gestion de projet, ...) qui répond à un besoin de l'entreprise fictive GSB.</p>		
Description des ressources documentaires, matérielles et logicielles utilisées⁶ <ul style="list-style-type: none"> - Environnement Local : Emulateur android studio et server web local WAMP, - Langages Frontend : Java, - Langage Backend : PHP, - Framework Frontend : Android Studio, - Bibliothèque API : Retrofit, - Tests : Android Studio Unit Test, - IDE : Vscodé et IntelliJ IDE, - Logiciel de gestion de la base de données : Phpmyadmin, - Versioning : Git depuis IntelliJ, Github pour le partage et la sauvegarde, - Gestion de projet : Excel pour le planning et la gestion des tâches, Paint pour les schémas, - Base de données : SQL, - Ressources documentaires : www.developer.android.com, www.stackoverflow.com, www.w3schools.com. 		
Modalités d'accès aux productions⁷ et à leur documentation⁸ <ul style="list-style-type: none"> - Code source : https://www.github.com/OscarPARDON/GSB_APP_MOBILE - Dossier et présentation : https://www.OscarPardon.com/projects/gsbNews 		

**ANNEXE 9-1-B : Fiche descriptive de réalisation professionnelle
(verso, éventuellement pages suivantes)****Épreuve E6 - Conception et développement d'applications (option SLAM)****Descriptif de la réalisation professionnelle, y compris les productions réalisées et schémas explicatifs**

GSB News est une application qui vise à faciliter la communication interne à l'entreprise GSB. L'application permet de partager des publications avec des images à tous les employés. Ces publications peuvent permettre d'informer, de sensibiliser, de divertir et bien d'autres choses en fonction du but recherché par les responsables. Le projet consistait à développer une application mobile ergonomique, structurée autour d'une interface utilisateur intuitive et d'un backend sécurisé et performant, les deux communiquant par API.

Pour accéder à l'application, tous les utilisateurs doivent passer par l'interface de connexion. Pour accéder à la page principale, l'utilisateur doit être inscrit par un admin. Une fois cela fait, il pourra se connecter avec son email et son mot de passe.

La page principale affiche toutes les publications ordonnées par pertinence. Un algorithme détermine la pertinence grâce à la date de publication du post et du nombre d'utilisateurs ayant aimé celui-ci. Cette page permet d'aimer et d'arrêter d'aimer un post.

La page de gestion des publications, accessible via le menu en bas de l'écran pour les utilisateurs en ayant l'accès, permet déjà d'accéder au formulaire de création d'une publication. Cette page permet aussi de modifier et de supprimer les publications accessibles en fonction du rôle.

La page de gestion des utilisateurs, accessible via le menu en bas de l'écran pour les utilisateurs en ayant l'accès, affiche dans un premier temps tous les utilisateurs actuels de l'application. Cette page permet aussi de créer, modifier et supprimer des utilisateurs.

L'application comporte donc de nombreuses fonctions dont l'accès varie en fonction du rôle de l'utilisateur. L'application offre 3 rôles ayant des niveaux d'accès différents.

Le premier rôle et le plus faible est le rôle d'utilisateur classique, ce rôle permet simplement de voir les publications et de les aimer.

Le rôle intermédiaire est le rôle de responsable ou manager, ce rôle permet, en plus d'avoir les mêmes fonctionnalités que les utilisateurs, de créer des publications ainsi que de modifier et supprimer ses propres publications uniquement.

Enfin le rôle le plus puissant est celui d'administrateur, il a accès à toutes les fonctionnalités. Il peut voir, aimer, modifier et supprimer toutes les publications, il peut ainsi jouer le rôle de modérateur si besoin. L'administrateur peut également créer ses propres publications. Enfin il a accès au module de gestion utilisateur qui lui permet de créer, modifier et supprimer les utilisateurs.

CONTEXTE

A / Présentation de Galaxy Swiss Bourdin (GSB)

Galaxy Swiss Bourdin (GSB) est un laboratoire pharmaceutique issu de la fusion, en 2009, de deux acteurs majeurs du secteur : l'américain Galaxy, spécialisé dans les traitements contre les maladies virales telles que le SIDA et les hépatites, et le conglomérat européen Swiss Bourdin, expert en médicaments d'usage courant.

Ce rapprochement a donné naissance à un leader de l'industrie pharmaceutique. Le siège administratif de Galaxy Swiss Bourdin Europe est basé à Paris, tandis que le siège mondial de la multinationale est situé à Philadelphie, aux États-Unis.

GSB s'appuie sur une force commerciale dédiée à l'information et au conseil auprès des professionnels de santé. En France métropolitaine (y compris la Corse), 480 visiteurs médicaux assurent ce rôle, auxquels s'ajoutent 60 représentants dans les départements et territoires d'outre-mer. Leur mission consiste à informer médecins, pharmaciens, infirmiers et autres prescripteurs sur les produits du laboratoire. Bien qu'ils ne réalisent pas directement de ventes, leur travail influence significativement la prescription des médicaments développés par GSB.

L'organisation de l'entreprise repose sur une structure hiérarchique régionale, regroupée en plusieurs grands secteurs géographiques (Sud, Nord, Paris-Centre, Antilles-Guyane, etc.), garantissant ainsi une couverture efficace du territoire.

Les principaux concurrents du laboratoire en France sont les groupes Sanofi, Viatris et enfin le groupe Bayer.

B / Contexte de la demande

Après la crise du Covid, l'état investit massivement dans le milieu pharmaceutique. C'est l'occasion pour l'entreprise Galaxy Swiss Bourdin de développer son activité en France. Dans ce cadre, l'entreprise souhaiterait améliorer la communication interne à l'entreprise.

Sur décision du responsable du service communication et de la direction, GSB souhaite donc acquérir un blog interne sous forme d'une application mobile pour permettre un accès à tout moment au produit pour les employés.

L'entreprise souhaite faire appel à un développeur indépendant en Freelance pour réaliser cette tâche.

Attentes :

- Faciliter le partage d'informations dans l'entreprise,
- Faciliter la sensibilisation des employés,
- Uniformiser le processus de communication interne,
- Offrir un support visuel pour la communication,
- Offrir la possibilité aux employés de donner leur avis à propos du contenu,
- Prioriser l'affichage de contenu pertinent.

Existant :

Jusqu'à présent, l'émission de contenu est gérée par le service communication, le service des ressources humaines et la direction. Le contenu est partagé principalement par email ou directement en personne via des supports physique (tracts, notes, ...) ou des interactions.

Le but de l'application est de remplacer les solutions existantes, en offrant un support de partage efficace et consultable n'importe où avec son smartphone professionnel. Ainsi le contenu sera regroupé à un seul endroit, sous un même format, de manière ergonomique et mobile.

Objectifs :

- Aucune impression de supports physique pour la communication interne,
- 100% du contenu de communication interne transite via GSB News,
- 100 utilisateurs quotidiens 1 mois après le lancement.

Cible :**Cible Principales :**

- Tous les employés,
- La direction,
- Le service des ressources humaines,
- Le service communication.

Cibles Secondaire :

- Les entreprises souhaitant améliorer leur communication interne.

C / Fonctionnalités et contenu attendus

Authentification :

L'application ne devra en aucun cas être accessible sans passer par la connexion. Une interface de connexion devra donc permettre aux utilisateurs d'accéder au contenu grâce à leur email et leur mot de passe. Seuls les utilisateurs ayant un compte créé par un administrateur pourront se connecter.

Page principale (flux de contenu) :

La page principale sur laquelle l'utilisateur arrive doit être accessible par tous les utilisateurs. Sur cette page, l'utilisateur devra pouvoir voir et faire défiler le contenu.

L'utilisateur devra également avoir un moyen de réagir au contenu (Like ou commentaire par exemple).

Enfin le contenu devra être ordonné en fonction de sa pertinence, celle-ci sera déterminée par l'ancienneté du post et de l'avis des utilisateurs. Par exemple un contenu récent avec des réactions positives de utilisateurs devra être placé en haut du flux, à l'inverse d'une publication ancienne avec peu de réaction.

Gestion de ses publications et modération :

Cette partie permettra aux responsables et aux administrateurs de gérer les publications. Tous les utilisateurs avec ces rôles pourront créer du contenu.

Les responsables pourront modifier et supprimer leurs propres publications uniquement.

Les administrateurs quant à eux pourront modifier et supprimer toutes les publications pour modérer le contenu si besoin.

Gestion des utilisateurs :

Cette partie destinée aux administrateurs devra comporter un moyen de gérer les utilisateurs.

Dans un premier temps, elle devra afficher la liste de tous les utilisateurs ayant actuellement accès à l'application.

Elle devra aussi permettre d'ajouter de nouveaux utilisateurs en fournissant le nom, le prénom, l'email, le mot de passe et le rôle de l'employé.

Prestations attendues :

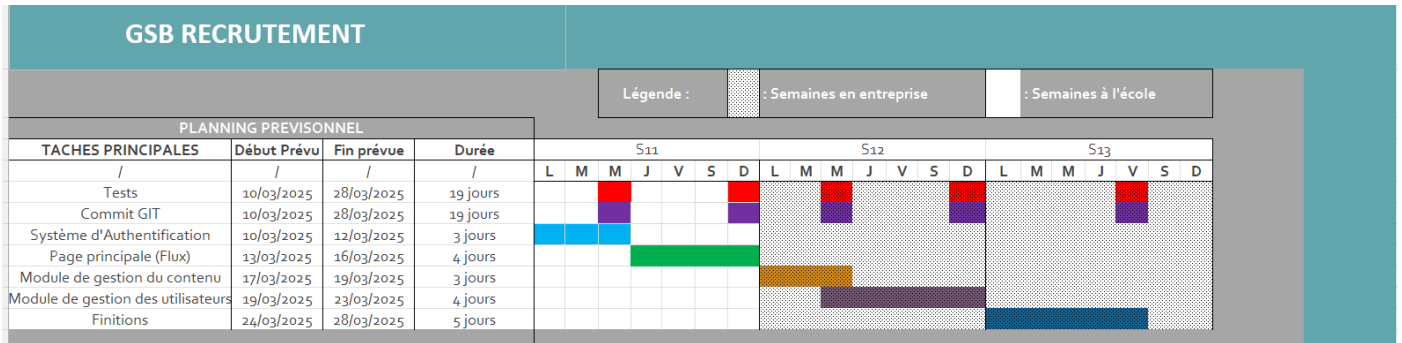
Le développement de la totalité de l'application sera effectué par PARDON Oscar en suivant les spécifications évoquées plus haut.

L'application devra être compatible avec les outils actuels de l'entreprise. L'application devra donc être accessible via des smartphones Android et le backend devra être implémentable sur un serveur

Debian pour la mise en production. Outre ces contraintes, il n'y a aucune directive sur les technologies à utiliser.

L'hébergement sera pris en charge par GSB pour le matériel et la mise en production.

D / Planning prévisionnel et budget

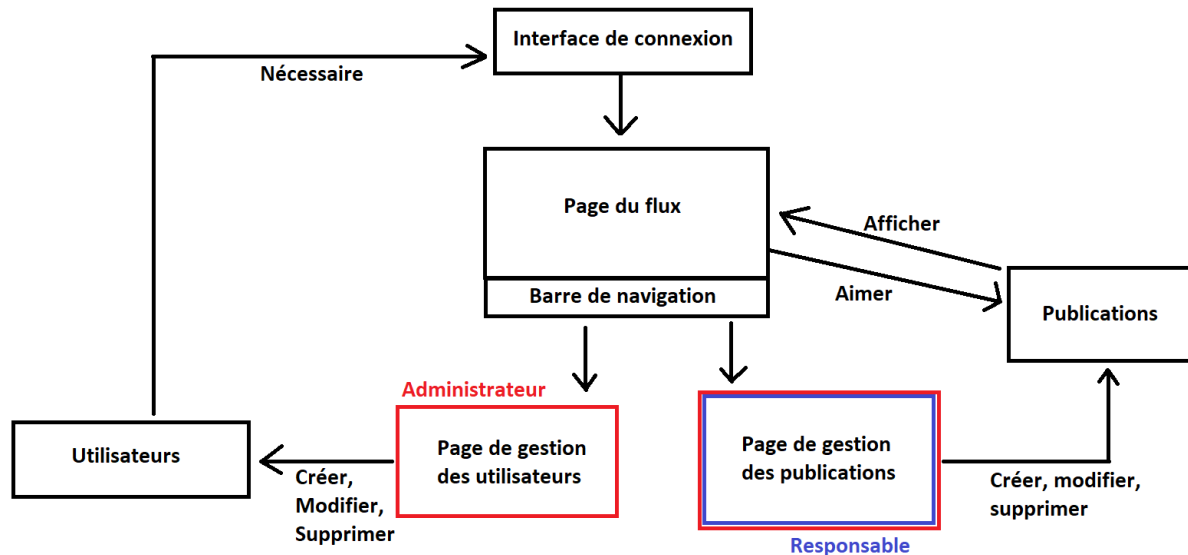


Le temps estimé de développement est de 1 mois, pour un cout de développement de 200€/jour ouvré, le budget du projet est estimé à environ 4 000 €.

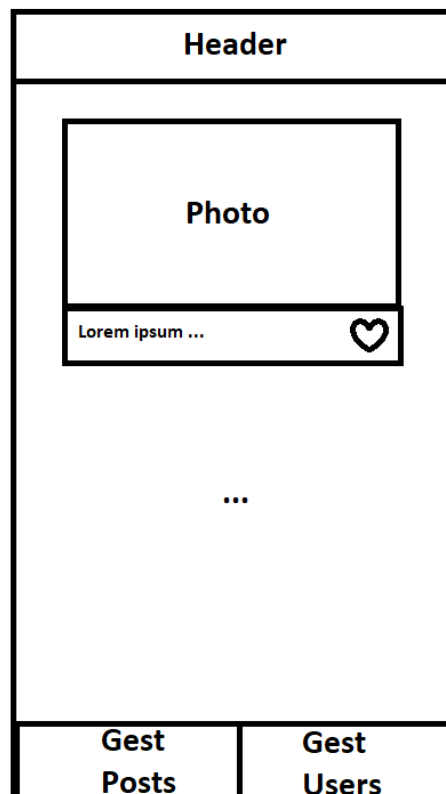
Préparation du projet :

A / Ebauches de l'application :

Un schéma de la structure de l'application a été effectué avant le début de la programmation :



De plus, avant la réalisation du « Layout » d'une page, une maquette était systématiquement dessinée. Voici un exemple de maquette pour la page principale :



B / Modèle conceptuel de données



C / Liste des tâches à réaliser :

Dans le cadre de la méthode AGILE, le projet a été découpé en tâches dont voici la liste :

Module	Tâche	Statut
Base de donnée	Créer les tables dans la base de donnée	En Attente
Authentification	Connexion	En Attente
Page principale	Affichage des publication	En Attente
	Aimer / Ne plus aimer les publications	En Attente
	Ordonner les publications par pertinence	En Attente
Gestion des publications	Formulaire de création d'une publication	En Attente
	Formulaire de modification d'une publication	En Attente
	Supprimer une publication	En Attente
Gestion des utilisateurs	Affichage de tous les utilisateurs	En Attente
	Formulaire de création d'un utilisateur	En Attente
	Formulaire de modification d'un utilisateur	En Attente
	Suppression d'un utilisateur	En Attente

Explications générales de la conception de l'application :

A / Fonctionnalités principales :

Authentification :

- Une interface de connexion permet de s'authentifier grâce à son email et son mot de passe. Il est obligatoire de passer par l'interface de connexion avant d'accéder à l'application.

Fonctionnalités de la page principale :

- Ceci est la première page sur laquelle l'utilisateur arrive après s'être connecté. Elle affiche le flux de contenu ordonné par pertinence (Ancienneté du post et avis des utilisateurs).
- Il est possible de d'aimer un post, et si l'utilisateur change d'avis il peut arrêter d'aimer un post. Aimer un post aura un impact sur la pertinence de celui-ci.
- Il est possible d'accéder aux pages de gestion des publications et de gestion des utilisateurs depuis la barre de navigation en bas de cette page.

Fonctionnalités de la page de gestion des publications :

- Cette page n'est accessible que par les responsables et les administrateurs. Si l'utilisateur possède le rôle de responsable, seules les publications qu'il a lui-même créées s'afficheront. En revanche si l'utilisateur possède un rôle d'administrateur, toutes les publications s'afficheront.
- Un bouton situé en bas de la page permet d'accéder au formulaire de création d'une publication.
- Une croix sur chaque publication vous permet de supprimer celles-ci.
- Un bouton illustré par un stylo présent sur chaque publication vous permet de modifier la publication ciblée avec un formulaire.

Fonctionnalités de la page de gestion des utilisateurs :

- Cette page n'est accessible que par les administrateurs. Elle contient une liste de tous les utilisateurs ayant actuellement accès à l'application.
- Un bouton en bas de la page permet d'accéder au formulaire de création d'un nouvel utilisateur.
- Une croix sur la ligne de chaque utilisateur permet de supprimer ceux-ci.
- Un bouton illustré par un stylo situé sur la ligne de chaque utilisateur, permet d'accéder au formulaire de modification de l'utilisateur.

B / Rôles :

L'application comporte trois rôles qui régissent l'accès aux différentes parties de celle-ci.

Le niveau le plus bas d'accès s'obtient avec le rôle de **l'utilisateur basique** qui a simplement accès au flux de contenu. Les utilisateurs peuvent également aimer les publications.

Le niveau d'accès intermédiaire s'obtient avec le rôle de **manager ou responsable** qui peut, en plus d'avoir accès au flux, créer ses propres publications qui viendront alimenter le flux. Ce rôle permet enfin de modifier et de supprimer ses propres publications.

Le niveau d'accès le plus haut s'obtient avec le rôle **d'administrateur**. Ce rôle permet en plus d'avoir accès au flux, de modérer (Supprimer et modifier) n'importe quelle publication. L'administrateur peut aussi créer ses propres publications qui viendront alimenter le flux. Pour finir ce rôle donne accès au module de gestion utilisateur, il peut donc créer, modifier et supprimer des utilisateurs.

Explications techniques de la conception de l'application :

A / Technologies utilisées

Environnement de développement :

- Framework Frontend : Android Studio (Java),
- Language Backend : PHP (OOP),
- IDE : IntelliJ (Jetbrains) & Visual Studio Code,
- Gestion de la base de données : PHP My Admin,
- Smartphone : Emulateur Android Studio,
- Server API : WAMP.

Base de données & API :

- Système de la Base de données : MySQL,
- Bibliothèque API : Retrofit,
- Test des API : Postman.

Gestion de projet :

- Versioning : Git & Github,
- Planning et gestion des tâches : Excel,
- Maquettes et schémas : Paint,

Tests :

- Tests unitaires : Junit dans Android Studio.

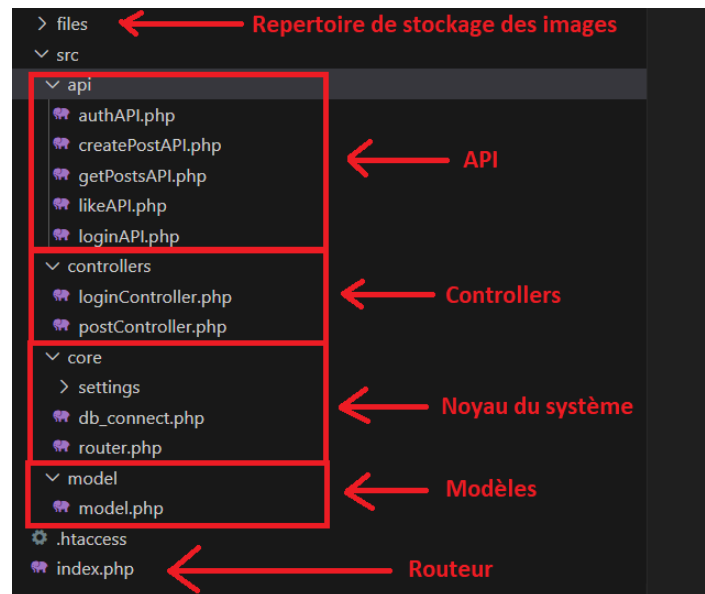
B / Arborescence et architecture de l'application

Backend :

Le backend de l'application est constitué sur une architecture API, contrôleur, modèle. Un routeur permet d'appeler la ressource demandée en fonction de l'URL.

Le fichier « .htaccess » permet de transformer l'URL en un URL facilement interprétable par le routeur. Par exemple http://192.168.0.0/GSB_MOBILE_API/api/login sera transformé en « http://192.168.0.0/GSB_MOBILE_API/index.php?route=login ».

Arborescence du Backend PHP :



Lorsqu'une requête est effectuée, le routeur appelle la ressource demandée qui va utiliser un contrôleur pour traiter la demande. Celui-ci va lui-même appeler le modèle pour récupérer les données dont il a besoin dans la base de données. Une réponse sera ensuite renvoyée sous forme de JSON.

Extrait du routeur :

```
<?php
header("Content-Type: application/json");
require 'src/core/Router.php';

$routeur = new Router();

$routeur->add('POST', 'login', function() {
    require_once 'src/api/loginAPI.php';
});

$routeur->add('POST', 'authVerif', function() {
    require_once 'src/api/authAPI.php';
});

$routeur->add('POST', 'getPosts', function() {
    require_once 'src/api/getPostsAPI.php';
});

$routeur->add('POST', 'createPost', function() {
    require_once 'src/api/createPostAPI.php';
});

$routeur->add('POST', 'like', function() {
    require_once 'src/api/likeAPI.php';
});

$routeur->dispatch();

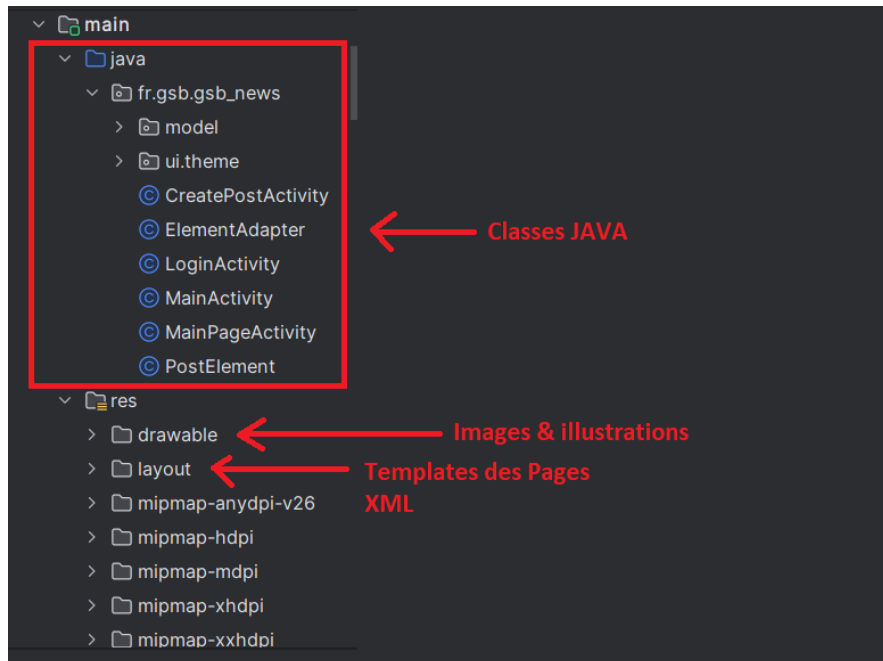
$pdo = null;

?>
```

Frontend :

L'architecture de l'application mobile est automatiquement créée par Android Studio. Elle est constituée de classes JAVA qui ont pour rôle de gérer les interactions avec le « layout » et le serveur Web. Les « layout » sont les pages affichées sur le smartphone de l'utilisateur.

Arborescence du Frontend Android Studio :



Les classes JAVA communiquent avec le serveur grâce à la bibliothèque API « Retrofit ».

Exemple de requête API « Retrofit » pour aimer un post :

```
interface ApiService {
    1 usage
    @FormUrlEncoded
    @POST("like")
    Call<ResponseBody> like(
        @Field("user_id") String userId,
        @Field("session_token") String sessionToken,
        @Field("post_id") int postId,
        @Field("like") int isLiked
    );
}

public void sendLike(int postId, int like) {
    // Vérifiez que le contexte n'est pas null
    if (context == null) return;

    SharedPreferences sharedPreferences = context.getSharedPreferences("app_prefs", Context.MODE_PRIVATE);
    String userId = sharedPreferences.getString("user_id", "null");
    String sessionToken = sharedPreferences.getString("session_token", "null");

    Retrofit retrofit = createRetrofitInstance(context.getString(R.string.api_url));
    ApiService apiService = retrofit.create(ApiService.class);

    Call<ResponseBody> call = apiService.like(userId, sessionToken, postId, like);
    call.enqueue(new retrofit2.Callback<ResponseBody>() {
```

C / Sécurité :

Authentication :

L'authentification est obligatoire pour accéder à l'application. Pour s'authentifier, il faut envoyer une requête au server Web avec son email et son mot de passe. Si les identifiants sont corrects, le server va créer une session avec une durée déterminée (Par défaut 2h) et renvoyer un jeton de session. Ce jeton associé à un utilisateur permet d'accéder aux autres fonctionnalités du server tant que celui-ci n'est pas expiré.

Exemple de session :

id	user_id	token	duration
33	2	724e86cb89c80dd50e47c778974b0ad7ef2e3e8e04c24dba40...	1742857519

La validité de la session est vérifiée côté Frontend avant d'afficher n'importe quelle page. Elle est aussi vérifiée systématiquement côté Backend avant de traiter une requête.

Méthode de vérification de l'authentification dans le modèle Utilisateur :

```
public function is_authenticated($user_id,$session_token) {  
    $session = (new Session())->get_session($user_id);  
    if($session && $session["duration"] > time() && $session["token"] == $session_token){  
        return ['status'=>200, 'authenticated'=>1];  
    }  
    return ['status'=>400, 'authenticated'=>0];  
}
```

Formulaires :

Les formulaires contiennent de nombreuses vérifications des données. Au niveau du frontend, l'utilisateur est encadré par des vérifications qui empêche l'envoi des données au server si les données ne sont pas conformes.

Mais surtout, les données sont aussi systématiquement vérifiées dans le backend avant d'être insérées dans la base de données.

Exemple de vérification des données au niveau du Backend :

```
$verif = 1;

if(!($description && strlen($description) <= 500)){
    $verif = 0;
}
if(!$image){
    $verif = 0;
}
if(!($user_id && (new User)->exists($user_id))){
    $verif = 0;
}
if($verif){
    $post = new Post;
    $post_id = $post->createPost($description,$user_id,$datetime);
    if($post_id){
        include("src/core/settings/settings.php");
        mkdir($POST_IMG_REPO . $post_id,0777,true);
        if (!file_put_contents($POST_IMG_REPO . $post_id . "/" . $post_id . ".png", $image_data))
            return ["status"=>500,"message"=>"Erreur lors de l'enregistrement de l'image"];
        return ["status"=>200,"success"=>1];
    }
    else{
        return ["status"=>500,"message"=>"Erreur lors de l'enregistrement du formulaire dans la ba"];
    }
}
```

Injection SQL :

L'utilisation de PDO permet une protection totale contre les injections SQL.

Hashage des mots de passe :

Les mots de passe sont évidemment hashés avant d'être stockés dans la base de données. Lors de la connexion, le mot de passe renseigné est hashé pour être comparé au hash présent dans la base de données.

D / Tests

Des tests unitaires ont été développés tout le long du développement pour éviter les régressions. Android Studio fournit une fonctionnalité de test unitaire implémentable de le répertoire Test.

Exemple de test unitaire dans pour tester la fonctionnalité d'aimer une publication :

```
public void testSendLikeSuccess() {
    // Préparation du mock pour l'appel API
    when(mockApiService.like(anyString(), anyString(), anyInt(), anyInt()))
        .thenReturn(mockCall);

    doAnswer(invocation -> {
        Callback<ResponseBody> callback = invocation.getArgument(index: 0);
        ResponseBody mockResponseBody = mock(ResponseBody.class);
        callback.onResponse(mockCall, Response.success(mockResponseBody));
        return null;
    }).when(mockCall).enqueue(any(Callback.class));

    // Exécution de la méthode à tester
    elementAdapter.sendLike(postId: 123, like: 1);

    // Vérifications
    verify(mockApiService).like(userId: "testUserId", sessionToken: "testSessionToken", postId: 123, isLiked: 1);
}
```

Bilan post-réalisation

A / Planning réel de la réalisation de l'application

PLANNING REEL				S11							S12							S13						
TACHES PRINCIPALES	Début Réel	Fin réelle	Durée réelle	L	M	M	J	V	S	D	L	M	M	J	V	S	D	L	M	M	J	V	S	D
/	/	/	/																					
Tests	10/03/2025	28/03/2025	19 jours																					
Commit GIT	10/03/2025	28/03/2025	19 jours																					
Système d'Authentification	10/03/2025	12/03/2025	3 jours																					
Page principale (Flux)	13/03/2025	18/03/2025	6 jours																					
Module de gestion du contenu	19/03/2025	23/03/2025	5 jours																					
Module de gestion des utilisateurs	24/03/2025	27/03/2025	4 jours																					
Finitions	28/03/2025	28/03/2025	1 jour																					

On constate des écarts avec le planning prévisionnel mais une marge de temps avait été prévue.

B / Versionning

Le versionning a été assuré durant toute la durée du développement notamment grâce l'implémentation de Git directement dans l'IDE IntelliJ. Git était connecté au répertoire Github « OscarPARDON/GSB_APP_MOBILE » dans lequel vous pouvez retrouver le code source.

C / Evolutions possibles

Compte tenu du temps de développement très limité (21 jours), l'application reste minimaliste et de nombreuses améliorations sont possible. Voici quelques exemples :

- Ajouter la possibilité de commenter les publications,
- Ajouter la possibilité de ne pas aimer une publication,
- Ajouter des notifications lors de l'arrivée de contenu,
- Ne charger qu'une partie des publications à la fois et non tout d'un coup,
- Améliorer la gestion de mot de passe des utilisateurs (Réinitialisation, mot de passe temporaire, ...),
- Ajouter une fonctionnalité de recherche pour les publications et les utilisateurs.

Documentation Utilisateur

Sommaire :

1. Connexion :

1.1 Se connecter

2. Page principale (Flux) :

2.1 Informations affichées

2.2 Aimer une publication

3. Gestion des publications :

3.1 Informations affichées

3.2 Créer une publication

3.3 Modifier une publication

3.4 Supprimer une publication

4. Gestion des utilisateurs :

4.1 Informations affichées

4.2 Créer un utilisateur

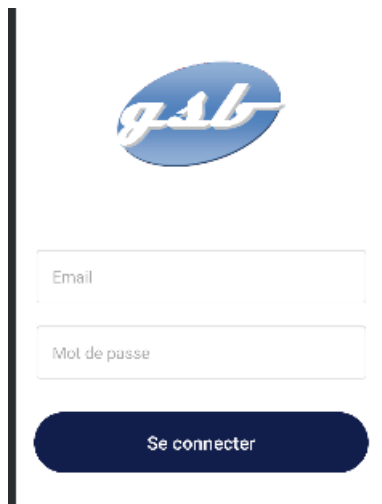
4.3 Modifier un utilisateur

4.4 Supprimer un utilisateur

1. Connexion

1.1 Se connecter

En arrivant sur l'application, une authentification vous sera demandée. Entrez votre email et votre mot de passe dans les champs correspondants et appuyez sur « Se Connecter ».

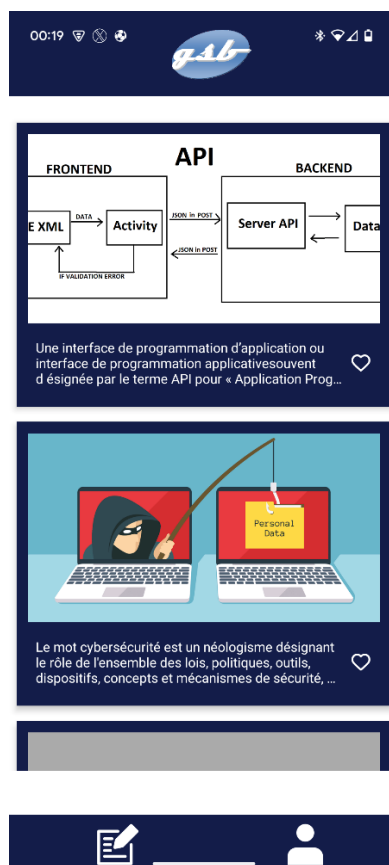


The login screen features the 'gsb' logo at the top. Below it are two input fields: 'Email' and 'Mot de passe' (Password). At the bottom is a dark blue button with the text 'Se connecter' (Log in).

2. Page principale (Flux) :

2.1 Informations affichées :

Cette page est la première sur laquelle vous arriverez une fois connecté. Elle affiche le flux de contenu.



2.2 Aimer une publication :

Pour aimer une publication, il suffit d'appuyer sur le cœur en bas à droite de la publication (Voir 2.1 Informations affichées pour illustration). Si vous ne souhaitez plus aimer une publication, appuyez à nouveau sur le cœur et il sera retiré.

3. Gestion des publications :

3.1 Informations affichées :

Cette page est accessible depuis la barre de navigation de la page principale (voir 2.1 pour illustration). Elle permet de gérer ses propres publications ou toutes les publications si l'utilisateur est un administrateur.

Vous pouvez y voir un flux de contenu comme sur la page principale, cependant les fonctionnalités ne sont pas les mêmes. Ici on ne trouve pas de cœur sur les publications mais plutôt une croix en haut à droite et un stylo en bas à droite pour chacune d'entre elles.

Un bouton « + » se situe dans une barre en bas de la page.

3.2 Créer une publication :

Pour créer une publication, cliquez sur le bouton « + » situé dans la barre en bas de la page. Vous serez renvoyé sur un formulaire. Renseignez ensuite une image et une description puis cliquez sur « Publier » pour envoyer votre publication dans le flux.



The screenshot shows the mobile app interface for creating a new publication. At the top, there is a dark blue header bar with the time '23:33' on the left, the 'gsb' logo in the center, and status icons (Bluetooth, Wi-Fi, battery) on the right. Below the header, there is a large grey rectangular area with a small image icon and the text 'Ajoutez une image'. Underneath this area is a dark blue button with a camera icon and the text 'Ajouter une Image', followed by the text 'Aucune image sélectionnée'. Below the button is a white rectangular text input field with the placeholder text 'De quoi souhaitez-vous parler ?'. At the bottom of the screen is a dark blue rounded rectangular button with the text 'Publier'.

3.3 Modifier une publication :

Pour modifier une publication, commencez par appuyer sur le stylo situé en bas à droite de la publication ciblée. Vous serez renvoyé sur un formulaire de modification sur lequel vous pourrez effectuer vos corrections. Une fois terminé, appuyez sur « Enregistrer » et les modifications seront prise en compte.

3.4 Supprimer une publication :

Pour supprimer une publication, il vous suffit de cliquer sur la croix en haut à droite de la publication ciblée. Ensuite confirmez si vous souhaitez poursuivre et la publication sera supprimée.

4. Gestion des utilisateurs :

4.1 Informations affichées :

Cette page est accessible depuis la barre de navigation de la page principale (voir 2.1 pour illustration) si vous êtes administrateur. Elle permet de gérer tous les utilisateurs.

Vous pouvez y voir une liste de tous les employés qui ont actuellement accès à l'application. A la fin de chaque ligne, vous trouverez une croix et un stylo.

Un bouton « + » se situe dans une barre en bas de la page.

4.2 Créer un utilisateur :

Pour créer une publication, cliquez sur le bouton « + » situé dans la barre en bas de la page. Vous serez renvoyé sur un formulaire. Renseignez les informations demandées puis cliquez sur « Créer » pour ajouter l'utilisateur. N'oubliez pas de communiquer son mot de passe à l'utilisateur concerné.

4.3 Modifier un utilisateur :

Pour modifier un utilisateur, commencez par appuyer sur le stylo situé à la fin de la ligne de l'utilisateur ciblé. Vous serez renvoyé sur un formulaire de modification sur lequel vous pourrez effectuer vos corrections. Une fois terminé, appuyez sur « Enregistrer » et les modifications seront prise en compte.

4.4 Supprimer un utilisateur :

Pour supprimer un utilisateur, il vous suffit de cliquer sur la croix à la fin de la ligne de l'utilisateur ciblé. Ensuite confirmez si vous souhaitez poursuivre et l'utilisateur sera supprimée.