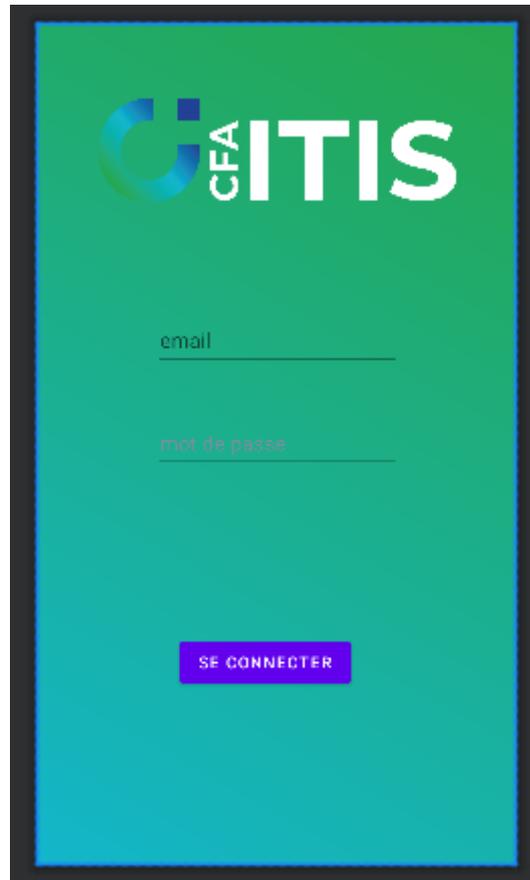


Documentation Technique



Dounia Sahli
BTS SIO

Table des matières

Dounia Sahli	1
BTS SIO	1
CONTEXTE	3
OUTILS UTILISE.....	4

CONTEXTE .

L'association BDE ITIS EVENTS à fait appelle à notre entreprise fictive « SDDEVOPS » afin de développer une application mobile pour la vie des étudiants du CFA.

Le contexte consiste à mettre en place une application mobile développée sous Android dédiée à l'association étudiante de l'école ITIS FORMATION afin de gérer la billetterie de chaque offre proposée pour les étudiants. Actuellement les billets sont délivrés manuellement. Le bureau des étudiants souhaite automatiser cette tâche.

Dans ce contexte, les sessions utilisateurs seront nécessaires pour la connexion au forum des étudiants (chaque utilisateur disposera d'une adresse mail scolaire et d'un mot de passe modifiable). Les étudiants pourront avoir accès aux différentes offres proposées par l'association étudiante avec une possibilité de paiement en ligne. Un espace administrateur permettra aux dirigeants de supprimer ou d'ajouter une nouvelle offre, supprimer, modifier ou ajouter un étudiant et de pouvoir modifier la page concernant la description du BDE ITIS EVENTS.

OUTILS UTILISE.

XAMP



Xamp est en mesure de lancer des sites web dynamiques qu'il soit connecté à Internet ou en local. Il peut être utilisé avec de simples navigateurs et contient un serveur Apache permettant de transférer des fichiers en ligne. Il intègre PhpMyAdmin, SGBD fonctionnant sur le langage Mysql.

JAVA



Java est un langage de programmation populaire et polyvalent. Il est utilisé pour développer une grande variété d'applications, allant des applications de bureau aux applications mobiles et aux systèmes embarqués. Java est connu pour sa portabilité, ce qui signifie que le code Java peut s'exécuter sur différentes plateformes sans avoir besoin d'être modifié. Il est également orienté objet, ce qui facilite la création de programmes modulaires et réutilisables. Java est largement utilisé dans l'industrie du développement logiciel et possède une vaste bibliothèque standard qui offre de nombreuses fonctionnalités prêtes à l'emploi.

Android Studio



Android Studio est un environnement de développement intégré (IDE) utilisé pour créer des applications Android. C'est l'outil principal pour les développeurs Android, offrant une interface conviviale et des fonctionnalités puissantes. Android Studio est basé sur IntelliJ IDEA et fournit des outils avancés tels que l'éditeur de code, le débogueur, l'émulateur d'appareil Android, la gestion des ressources, le profilage des performances, etc. Il facilite la création d'applications Android en fournissant une structure de projet, des modèles de code et une intégration étroite avec les outils de développement Android tels que le SDK Android et Gradle.

PHP



PHP (Hypertext Preprocessor) est un langage de script côté serveur utilisé principalement pour le développement web. Il est largement utilisé pour créer des sites web dynamiques et des applications web interactives. PHP est intégré directement dans le code HTML et peut se connecter aux bases de données pour récupérer et manipuler des données. Il offre une syntaxe simple et flexible, ce qui en fait un langage populaire parmi les développeurs. PHP est également open source, ce qui signifie qu'il est gratuit et dispose d'une vaste communauté de développeurs qui contribuent à son amélioration et à son évolution.

Volley

Volley est une bibliothèque de réseau Android développée par Google qui facilite les requêtes réseau et la gestion des réponses dans les applications Android. Elle offre une interface simple et concise pour effectuer des requêtes HTTP, gérer les files d'attente de requêtes, effectuer des demandes asynchrones et traiter les réponses en utilisant des rappels (callbacks). Volley est conçue pour améliorer les performances en optimisant la gestion du réseau, en offrant la mise en cache automatique des réponses et en prenant en charge la parallélisation des requêtes. Elle est largement utilisée pour récupérer des données à partir d'API web, effectuer des téléchargements d'images et d'autres tâches de communication réseau dans les applications Android.

Gson

Gson est une bibliothèque de sérialisation et de désérialisation JSON dans Android Studio. Développée par Google, elle permet de convertir des objets Java en format JSON (sérialisation) et de convertir des données JSON en objets Java correspondants (désérialisation). Gson offre une interface simple pour travailler avec JSON, en permettant la conversion automatique des propriétés d'objets Java en clés JSON et vice versa. Elle facilite ainsi l'échange de données entre une application Android et un serveur web, en convertissant les données dans un format standardisé et facilement manipulable. Gson prend également en charge la gestion des types complexes, tels que les listes, les tableaux et les objets imbriqués, offrant une solution pratique pour manipuler des données JSON dans le développement Android.

Étapes pour se connecter

- Créer le script php qui permet de se connecter en utilisant PDO.

```
<?php

header('content-type: text/html; charset=utf_8');

require "hpdDbConnect.php";

if(isset($_GET['email']) && isset($_GET['mdp']))
{
    // Initialize Variable
    $result='';
    $email = $_GET['email'];
    $mdp = $_GET['mdp'];
    $query = $bdd->prepare("SELECT * from users where email = :email and password = :password");

    $query->execute(array(':email'=>$email, ':password'=>$mdp));

while($row=$query->fetch(PDO::FETCH_ASSOC)){

    $result= $row ;

}
if (!empty($result))

    echo json_encode($result);

else

    echo "null";

}

?>
```

Pour exécuter ce script de connexion PHP avec Volley dans Android Studio en utilisant la méthode GET, nous pouvons suivre les étapes suivantes :

- On ajoute la dépendance Volley au projet Android Studio en ajoutant la ligne suivante dans le fichier build.gradle :

```
implementation 'com.android.volley:volley:1.2.1'
```

- Dans le code Java, on crée une fonction (« getData() ») et on utilise la classe RequestQueue pour gérer les requêtes Volley. Il est possible de le faire dans l'activité principale ou dans une classe dédiée :

```
mRequestQueue = Volley.newRequestQueue(MainActivity.this);
```

Il faut utiliser la classe `StringRequest` pour envoyer la requête GET au script PHP et recevoir la réponse. Voici un exemple du code :

```
String url = URL_Auth+"?email="+email.getText()+"&"+mdp.getText();
```

```
public void getData(String url){
//RequestQueue initialized
mRequestQueue = Volley.newRequestQueue(MainActivity.this);
Log.i("Response :", "response.toString()");
//String Request initialized
mStringRequest = new StringRequest(Request.Method.GET, url, new
Response.Listener<String>() {
@Override
public void onResponse(String response) {

        Gson gson = new Gson();
user = gson.fromJson(response, User.class);
//if( response.equals("null"))
Log.i("Response1 :", user.getEmail());
//display the response on screen

}
}, new Response.ErrorListener() {
@Override
public void onErrorResponse(VolleyError error) {

Toast.makeText(MainActivity.this,error.toString(),Toast.LENGTH_SHORT).show()
;
}
});
mRequestQueue.add(mStringRequest); }
```

- On ajoute la requête à la file d'attente de Volley

```
mRequestQueue.add(mStringRequest);
```

Il est possible de gérer la réponse du serveur dans la méthode `onResponse()` en fonction du format de données renvoyé par le script PHP. Si le script renvoie du JSON, nous pouvons utiliser la bibliothèque Gson pour convertir la réponse JSON en objets Java.

- Il ne faut pas oublier d'ajouter les autorisations d'accès à Internet dans le fichier `AndroidManifest.xml` :

```
<uses-permission android:name="android.permission.INTERNET" />
```

Ceci est une implémentation de base pour exécuter le script PHP avec Volley dans Android Studio. Nous pouvons l'adapter en fonction de nos besoins spécifiques et ajouter des fonctionnalités supplémentaires comme la gestion des erreurs et la sécurisation des données

Récupération des données d'attribution :

1)Créer le script php qui permet de récupérer les données de la table offre dans la base de données.

```
<?php

header('content-type: text/html; charset=utf_8');

require "hpddbConnect.php";

// $query = $bdd->prepare("SELECT a.id, a.titre, a.filiere, a.date_debut, a.duree,
a.adresse, a.ville, a.code_postal, a.telephone, a.description, a.date_ajout,
a.id_partenaire, a.id_user , u.id, u.nom, u.prenom, u.email, e.id, e.nom, e.id_img
from ajout_offre a, users u, entreprise e where a.id_partenaire = e.id");
$query = $bdd->prepare("SELECT `description`, `nb_place`, `lieu`, `categorie_id`,
`id_offres`, `prix`, date, libelle FROM `offres` , categorie WHERE
offres.categorie_id = categorie.id ");

$query->execute();
$offreData = array();
$allOffres = array();
// var_dump($query);
while ($row = $query->fetch(PDO::FETCH_ASSOC)) {

    $description = $row['description'];
    $nb_place = $row['nb_place'];
    $lieu = $row['lieu'];
    $categorie_id= $row['categorie_id'];
    $date= $row['date'];
    $id_offres = $row['id_offres'];
    $prix = $row['prix'];
    $libelle = $row['libelle'];

    $offreData = array("offre" => ["id" => $id_offres, "description" =>
    $description, "categorie" => ["id"=> $categorie_id, "libelle"=>$libelle], "lieu" =>
    $lieu, "prix" => $prix, "nb_place"=>$nb_place,"date"=>$date]); // "description"=>
    $description]);
    $allOffres['resultOffres'][] = $offreData;
}
echo json_encode($allOffres);

?>
```

Voici la représentation UML de la classe offre :

```
-----  
|   Offre   |  
-----  
| - id_offres: int   |  
| - description: String|  
| - date: String    |  
| - nb_place: String |  
| - lieu: String    |  
| - categorie_id: Categorie |  
| - prix: float     |  
-----  
| + Offre(id_offres: int, description: String, nb_place: String, lieu: String, categorie_id: Categorie, prix:  
float, date: String) |  
| + getDate(): String |  
| + setDate(date: String): void |  
| + getId_offres(): int |  
| + setId_offres(id_offres: int): void |  
| + setDescription(description: String): void |  
| + setNb_place(nb_place: String): void |  
| + setLieu(lieu: String): void |  
| + setCategorie_id(categorie_id: Categorie): void |  
| + setPrix(prix: float): void |  
| + getDescription(): String |  
| + getNb_place(): String |  
| + getLieu(): String |  
| + getCategorie_id(): Categorie |  
| + getPrix(): float |  
-----
```

En exécutant la requête PHP du fichier offre :

<http://localhost/android/selectOffre.php>

On obtient ce résultat qui est un tableau d'offre sous format Json :

```
{"resultOffres":[{"offre":{"id":"1","description":"Embarquez pour un joyeux tour du monde avec \"it's a small world\", d\u00e9collez pour l'espace avec Star Wars Hyperspace Mountain, croisez le fer dans Pirates of the Caribbean, ou montez dans l'un des wagonnets fous de Big Thunder Mountain, et bien d'autres !","categorie":{"id":"1","libelle":"1"},"lieu":"Chessy","prix":"50","nb_place":"10","date":"2023-08-07"}},{offre":{"id":"2","description":"Profitez d'une exp\u00e9rience cin\u00e9matographique inoubliable avec notre offre exclusive Gaumont Path\u00e9 ! Plongez dans l'univers captivant du grand \u00e9cran et laissez-vous emporter par des films \u00e0 couper le souffle!","categorie":{"id":"1","libelle":"1"},"lieu":"Essone","prix":"6","nb_place":"10","date":"2023-08-16"}},{offre":{"id":"3","description":"D\u00e9couvrez la beaut\u00e9 de la nature sauvage et partez \u00e0 l'aventure avec notre offre sp\u00e9ciale pour le Zoo de Thoiry !","categorie":{"id":"1","libelle":"1"},"lieu":"Thoiry","prix":"23","nb_place":"14","date":"2023-06-20"}}]}
```