**Definición de Servicios**

Para nuestra aplicación hemos definido los siguientes servicios:

|  |  |
| --- | --- |
| Titulo | Obtener Restaurantes, lista todos u obtiene un restaurante. |
| URL | api/restaurantes |
| Método | GET |
| Parámetros del URL | Opcionales: idRestaurante (para devolver un restaurante específico) |
| Parámetros de datos |  |
| Respuesta Exitosa | Código: 200 (HTTP Response)  Contenido: { data:{ id: [int], nombre: [string], idDistrito: [int], latitude: [decimal(11,8)], longitude: [decimal(11,8)], descripcion: [string], puntuacionTotal: [decimal(10,2)] } }  Ejemplo: { data: { id: 2, nombre: “Rustica”, idDistrito: 2, latitud: -77.0829897, longitud: -11.4897042, descripcion: “Restaurante con 21 años de creación”, puntuacionTotal: 8.3, estado: 1 } } |
| Respuesta de Error | Código: 401 UNAUTHORIZED  Contenido: {error: “Obtener Restaurantes”} |
| Ejemplo de Invocación | $.ajax({ url: " api/restaurantes", dataType: "json", type : "GET", success : function(r) { console.log(r); }}); |
| Notas |  |

|  |  |
| --- | --- |
| Titulo | Crear Restaurante, crea un restaurante. |
| URL | api/restaurantes/create |
| Método | POST |
| Parámetros del URL |  |
| Parámetros de datos | { nombre: [string], idDistrito : [int], latitud : [decimal(11,8)] , longitud : [decimal(11,8)], descripcion: [string], puntuacionTotal: [decimal(10,2)] }}  Ejemplo: { data : { nombre: “La Ultima Cena”, idDistrito : 6, latitud: -77.0829897, longitud: -11.4897042, descripcion: “restaurant bien bueno”, puntuacionTotal: 5.3 } |
| Respuesta Exitosa | Código: 200 (HTTP Response)  { data : { id: [int], nombre: [string], idDistrito : [int], latitud : [decimal(11,8)] , longitud : [decimal(11,8)], descripcion: [string], puntuacionTotal: [decimal(10,2)] }}  Ejemplo: { data : { id: 16, nombre: “La Ultima Cena”, idDistrito : 6, latitud: -77.0829897, longitud: -11.4897042, descripcion: “restaurant bien bueno”, puntuacionTotal: 5.3 }} |
| Respuesta de Error | Código: 401 UNAUTHORIZED  Contenido: {error: “Crear Restaurante} |
| Ejemplo de Invocación | $.ajax({ url: " api/restaurantes/create", dataType: "json", type : "POST", success : function(r) { console.log(r); }}); |
| Notas |  |

|  |  |
| --- | --- |
| Titulo | Restaurantes por Categoría, obtiene todos los restaurantes de una categoría. |
| URL | api/categorías/{id}/restaurantes |
| Método | GET |
| Parámetros del URL |  |
| Parámetros de datos |  |
| Respuesta Exitosa | Código: 200 (HTTP Response)  { data : { id: [int], nombre: [string], idDistrito : [int], latitud : [decimal(11,8)] , longitud : [decimal(11,8)], descripcion: [string], puntuacionTotal: [decimal(10,2)], estado: [int] } … }  Ejemplo: { data : { id: 16, nombre: “La Ultima Cena”, idDistrito : 6, latitud: -77.0829897, longitud: -11.4897042, descripcion: “restaurant bien bueno”, puntuacionTotal: 5.3, estado: 1 } … } |
| Respuesta de Error | Código: 401 UNAUTHORIZED  Contenido: {error: “Obtener Restaurantes por Categoria” } |
| Ejemplo de Invocación | $.ajax({ url: "api/categorías/8/restaurantes ", dataType: "json", type : "GET", success : function(r) { console.log(r); }}); |
| Notas |  |

|  |  |
| --- | --- |
| Titulo | Crear Recomendación, crea una nueva recomendación para un restaurante. |
| URL | api/recomendaciones/create |
| Método | POST |
| Parámetros del URL |  |
| Parámetros de datos | { comentario: [string], puntuación: [int], idUsuario : [int], idRestaurante: [int] , idReceptor: [int] }  Ejemplo: { comentario: “Este restaurante es casi perfecto!”, puntuación: 9, idUsuario : 26, idRestaurante: 14 , idReceptor: 9 } |
| Respuesta Exitosa | Código: 200 (HTTP Response)  { data : { id: [int], comentario: [string], puntuación: [int] idUsuario : [int], idRestaurante: [int] , idReceptor: [int] } }  Ejemplo: { data : { id: 32, comentario: “Este restaurante es casi perfecto!”, puntuación: 9, idUsuario : 26, idRestaurante: 14 , idReceptor: 9 } } |
| Respuesta de Error | Código: 401 UNAUTHORIZED  Contenido: {error: “Crear Recomendacion” } |
| Ejemplo de Invocación | $.ajax({ url: " api/recomendaciones/create", dataType: "json", type : "POST", success : function(r) { console.log(r); }}); |
| Notas |  |

|  |  |
| --- | --- |
| Titulo | Obtener Recomendación, obtiene las recomendaciones hechas a un usuario. |
| URL | api/recomendaciones |
| Método | GET |
| Parámetros del URL | Opcional: idUsuario (para devolver una lista de recomendaciones hechas a un usuario específico) |
| Parámetros de datos |  |
| Respuesta Exitosa | Código: 200 (HTTP Response)  { data : { id: [int], comentario: [string], puntuación: [int] idUsuario : [int], idRestaurante: [int] , idReceptor: [int], estado: [int] } … }  Ejemplo: { data : { id: 32, comentario: “Este restaurante es casi perfecto!”, puntuación: 9, idUsuario : 26, idRestaurante: 14 , idReceptor: 9, estado: 1 } … } |
| Respuesta de Error | Código: 401 UNAUTHORIZED  Contenido: {error: “Obtener Recomendación” } |
| Ejemplo de Invocación | $.ajax({ url: " api/recomendaciones ", dataType: "json", type : "GET", success : function(r) { console.log(r); }}); |
| Notas |  |

|  |  |
| --- | --- |
| Titulo | Crear Usuario, crea un usuario nuevo. |
| URL | api/usuarios/create |
| Método | POST |
| Parámetros del URL |  |
| Parámetros de datos | { nombres: [string], apellidos: [string] username : [string], emai: [string] , facebook\_id: [int], password: [string], idDistrito: [int], is\_admin: [int] }  Ejemplo: { nombres: “Juan”, apellidos: “Perez” username : “jperez”, emai: “jperez25@gmail.com” , facebook\_id: 23, password: “qwerty”, idDistrito: 12, is\_admin: 0 } |
| Respuesta Exitosa | Código: 200 (HTTP Response)  { data : { id: [int], nombres: [string], apellidos: [string] username : [string], emai: [string] , facebook\_id: [int], password: [string], idDistrito: [int], is\_admin: [int] } }  Ejemplo: { data : { id: 25, nombres: “Juan”, apellidos: “Perez” username : “jperez”, emai: “jperez25@gmail.com” , facebook\_id: 23, password: “qwerty”, idDistrito: 12, is\_admin: 0 } } |
| Respuesta de Error | Código: 401 UNAUTHORIZED  Contenido: {error: “Crear Usuario” } |
| Ejemplo de Invocación | $.ajax({ url: "api/usuarios/create ", dataType: "json", type : "GET", success : function(r) { console.log(r); }}); |
| Notas |  |

|  |  |
| --- | --- |
| Titulo | Actualizar Usuario, actualiza un usuario existente. |
| URL | api/usuarios/{id} |
| Método | PUT |
| Parámetros del URL |  |
| Parámetros de datos | { nombres: [string], apellidos: [string] username : [string], emai: [string] , facebook\_id: [int], password: [string], idDistrito: [int], is\_admin: [int] }  Ejemplo: { nombres: “JuanEditado”, apellidos: “PerezEditado” username : “jperez”, emai: “jperez25@gmail.com” , facebook\_id: 23, password: “qwerty2”, idDistrito: 12, is\_admin: 0 } |
| Respuesta Exitosa | Código: 200 (HTTP Response)  { data : { id: [int], nombres: [string], apellidos: [string] username : [string], emai: [string] , facebook\_id: [int], password: [string], idDistrito: [int], is\_admin: [int] } }  Ejemplo: { data : { id: 25, nombres: “JuanEditado”, apellidos: “PerezEditado” username : “jperez”, emai: “jperez25@gmail.com” , facebook\_id: 23, password: “qwerty2”, idDistrito: 12, is\_admin: 0 } } |
| Respuesta de Error | Código: 401 UNAUTHORIZED  Contenido: {error: “Actualizar Usuario” } |
| Ejemplo de Invocación | $.ajax({ url: "api/usuarios/2 ", dataType: "json", type : "PUT", success : function(r) { console.log(r); }}); |
| Notas |  |

|  |  |
| --- | --- |
| Titulo | Eliminar Usuario, elimina un usuario existente. |
| URL | api/usuarios/{id} |
| Método | DELETE |
| Parámetros del URL |  |
| Parámetros de datos |  |
| Respuesta Exitosa | Código: 200 (HTTP Response)  { data : { id: [int], message: [string]} }  Ejemplo: { data : { id: 25, message: “soft deleted” } } |
| Respuesta de Error | Código: 401 UNAUTHORIZED  Contenido: {error: “Eliminar Usuario” } |
| Ejemplo de Invocación | $.ajax({ url: "api/usuarios/3", dataType: "json", type : "DELETE", success : function(r) { console.log(r); }}); |
| Notas |  |

|  |  |
| --- | --- |
| Titulo | Obtener Usuarios, obtiene todos o un usuario específico. |
| URL | api/usuarios/ |
| Método | GET |
| Parámetros del URL | Opcional: idUsuario (para devolver un usuario específico) |
| Parámetros de datos |  |
| Respuesta Exitosa | Código: 200 (HTTP Response)  { data : { id: [int], nombres: [string], apellidos: [string] username : [string], emai: [string] , facebook\_id: [int], password: [string], idDistrito: [int], is\_admin: [int], estado: [int] } … }  Ejemplo: { data : { id: 25, nombres: “Juan”, apellidos: “Perez” username : “jperez”, emai: “jperez25@gmail.com” , facebook\_id: 23, password: “qwerty”, idDistrito: 12, is\_admin: 0, estado: 1 } … } |
| Respuesta de Error | Código: 401 UNAUTHORIZED  Contenido: {error: “Obtener Usuario” } |
| Ejemplo de Invocación | $.ajax({ url: "api/usuarios ", dataType: "json", type : "GET", success : function(r) { console.log(r); }}); |
| Notas |  |

|  |  |
| --- | --- |
| Titulo | Obtener Categorías, obtiene todas o una categoría específica. |
| URL | api/categorias |
| Método | GET |
| Parámetros del URL | Opcional: idCategoria (para devolver una categoría específica) |
| Parámetros de datos |  |
| Respuesta Exitosa | Código: 200 (HTTP Response)  { data : { id: [int], nombre: [string], descripcion: [string], estado: [int] } … }  Ejemplo: { data: { id: 7, nombre: “criolla”, descripcion: “comida peruana”, estado: 1 } … } |
| Respuesta de Error | Código: 401 UNAUTHORIZED  Contenido: {error: “Obtener Categoria” } |
| Ejemplo de Invocación | $.ajax({ url: "api/categorias ", dataType: "json", type : "GET", success : function(r) { console.log(r); }}); |
| Notas |  |

|  |  |
| --- | --- |
| Titulo | Obtener Preferencias, obtiene todas las preferencias de un usuario. |
| URL | api/preferencias |
| Método | GET |
| Parámetros del URL | Obligatorio: idUser (para obtener las preferencias del usuario con ese código) |
| Parámetros de datos |  |
| Respuesta Exitosa | Código: 200 (HTTP Response)  { data : { id: [int], idUsuario: [int], idCategoria: [int], estado: [int] } … }  Ejemplo: { data: { id: 7, idUsuario: 12, idCategoria: 8, estado: 1 } … } |
| Respuesta de Error | Código: 401 UNAUTHORIZED  Contenido: {error: “Obtener Preferencias” } |
| Ejemplo de Invocación | $.ajax({ url: "api/preferencias ", dataType: "json", type : "GET", success : function(r) { console.log(r); }}); |
| Notas |  |

|  |  |
| --- | --- |
| Titulo | Crear Preferencia, crea una preferencia para un usuario. |
| URL | api/preferencias/create |
| Método | POST |
| Parámetros del URL |  |
| Parámetros de datos | { idUsuario: [int], idCategoria: [int] }  Ejemplo: { idUsuario: 18, idCategoria: 6 } |
| Respuesta Exitosa | Código: 200 (HTTP Response)  { data : { id: [int], idUsuario: [int], idCategoria: [int], estado: [int] } }  Ejemplo: { data: { id: 9, idUsuario: 14, idCategoria: 7, estado: 1 } } |
| Respuesta de Error | Código: 401 UNAUTHORIZED  Contenido: {error: “Crear Preferencia” } |
| Ejemplo de Invocación | $.ajax({ url: "api/preferencias/create ", dataType: "json", type : "POST", success : function(r) { console.log(r); }}); |
| Notas |  |

|  |  |
| --- | --- |
| Titulo | Eliminar Preferencia, elimina una preferencia existente para un usuario. |
| URL | api/preferencias |
| Método | DELETE |
| Parámetros del URL |  |
| Parámetros de datos |  |
| Respuesta Exitosa | Código: 200 (HTTP Response)  { data : { id: [int], message: [string]} }  Ejemplo: { data : { id: 25, message: “soft deleted” } } |
| Respuesta de Error | Código: 401 UNAUTHORIZED  Contenido: {error: “Eliminar Preferencia” } |
| Ejemplo de Invocación | $.ajax({ url: "api/preferencias ", dataType: "json", type : "DELETE", success : function(r) { console.log(r); }}); |
| Notas |  |

|  |  |
| --- | --- |
| Titulo | Login, verifica las credenciales del usuario. |
| URL | api/usuario/login |
| Método | POST |
| Parámetros del URL |  |
| Parámetros de datos | { username: [string], password: [string] }  Ejemplo: { username: juanito123, password: contra123 } |
| Respuesta Exitosa | Código: 200 (HTTP Response)  { data : { id: [int], nombres: [string], apellidos: [string] username : [string], emai: [string] , facebook\_id: [int], password: [string], idDistrito: [int], is\_admin: [int] } }  Ejemplo: { data : { id: 25, nombres: “JuanEditado”, apellidos: “PerezEditado” username : “jperez”, emai: “jperez25@gmail.com” , facebook\_id: 23, password: “qwerty2”, idDistrito: 12, is\_admin: 0 } } |
| Respuesta de Error | Código: 401 UNAUTHORIZED  Contenido: {error: “Usuario no identificado” } |
| Ejemplo de Invocación | $.ajax({ url: "api/usuario/login ", dataType: "json", type : "POST", success : function(r) { console.log(r); }}); |
| Notas |  |

|  |  |
| --- | --- |
| Titulo | Crear foto, sube una foto al servidor de Flickr para que sea usada por la aplicación. |
| URL | api/fotos/create |
| Método | POST |
| Parámetros del URL |  |
| Parámetros de datos | { nombre: [string], foto: [file] }  Ejemplo: { nombre: myRestaurantPhoto, file: imagen.jpg } |
| Respuesta Exitosa | Código: 200 (HTTP Response)  { data : { id: [int], nombre: [string], url: [string], formato: [string], estado: [int] } }  Ejemplo: { data : { id: 3, nombre: myRestaurantPhoto, url: , formato: JPG, estado: 1 } } |
| Respuesta de Error | Código: 401 UNAUTHORIZED  Contenido: {error: “Crear foto” } |
| Ejemplo de Invocación | $.ajax({ url: "api/ fotos/create", dataType: "json", type : "POST", success : function(r) { console.log(r); }}); |
| Notas |  |

**Facebook SDK for Android**

El SDK de Facebook para Android es la manera más fácil de integrar nuestra aplicación RestaurApp con Facebook. Este SDK permitirá a nuestra aplicación:

* **Facebook Login**

Permite la autenticación con credenciales de Facebook. Permitirá a la aplicación usar el siguiente servicio:

|  |  |
| --- | --- |
| Titulo | Facebook Login, verifica las credenciales del usuario utilizando su cuenta de facebook. |
| URL |  |
| Método |  |
| Parámetros del URL |  |
| Parámetros de datos | Permisos de la aplicación [String]  loginButton=(LoginButton)view.findViewById(R.id.login\_button);  loginButton.setReadPermissions("user\_friends"); |
| Respuesta Exitosa | Se ejecutará la función onSuccess(LoginResult loginResult) |
| Respuesta de Error | Se ejecutará la función onError(FacebookException expception) |
| Ejemplo de Invocación | Una vez configurado el SDK, se llamará al login de Facebook mediante un botón:  <com.facebook.login.widget.LoginButton  android:id="@+id/login\_button"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_gravity="center\_horizontal"  android:layout\_marginTop="30dp"  android:layout\_marginBottom="30dp" /> |
| Notas | Revisar la documentación official del login con Facebook utilizando Facebook SDK for Android: <https://developers.facebook.com/docs/facebook-login/android/v2.3> |

* **Compartir y enviar diálogos**

Las personas pueden compartir contenido de la aplicación a Facebook. Permitirá a la aplicación usar el siguiente servicio:

|  |  |
| --- | --- |
| Titulo | Recomendar Restaurante, permite al usuario compartir en Facebook un restaurante. |
| URL |  |
| Método |  |
| Parámetros del URL |  |
| Parámetros de datos | * contentURL, el link a ser compartido * contentTitle, representa el título del contenido del link * imageURL, la URL o imágen miniatura que aparecerá en el post * contentDescription, 2 a 4 oraciones que describan el contenido   Ejemplo:  ShareLinkContent content = new ShareLinkContent.Builder() .setContentUrl(Uri.parse("https://developers.facebook.com")).build(); |
| Respuesta Exitosa |  |
| Respuesta de Error |  |
| Ejemplo de Invocación | ShareButton shareButton = (ShareButton)findViewById(R.id.fb\_share\_button);  shareButton.setShareContent(content); |
| Notas | Revisar la documentación official de compartir con Facebook utilizando Facebook SDK for Android: <https://developers.facebook.com/docs/sharing/android>  Hay varias formas de compartir, en la documentación están los ejemplos. |

Reference: <https://developers.facebook.com/docs/android>