

# Documentación protocolo REST - MensajerO

## Recursos disponibles

- Login
- Conversacion
- Usuario

*Formato de contenido: json*

---

## Recurso - Login

**Acción:** loguear usuario en el sistema

**URI:** server/login

**Método:** POST

**Body:** { "id": <str:usuario>,  
          "password": <str:password>  
          }

*Restricciones sobre identificador de usuario: alfanumérico, máximo 12 caracteres.*

### Reply

En caso de usuario y contraseña válidos:

- **Response code:** 201 Created
- **Body:** { "token": <str:login\_token> }

En caso de usuario o contraseña inválido:

- **Response code:** 401 Unauthorized request
- 

## Recurso - Usuario

**Acción:** solicitar lista de usuarios con su estado correspondiente, incluyendo si tiene mensajes sin leer

**URI:** server/usuarios

**Método:** GET

**Params:** r\_user=<str:appuser\_id>&token=<str:login\_token>

### Reply

En caso de r\_user/token inválido:

- **Response code:** 401 Unauthorized request

En case de token valido:

- **Response code:** 200 OK

- **Body:** [ { "id": "pepe", "nombre": "Pepito", "estado": "conectado" }, { "id": "carlos", "nombre": "Carlos", "estado": "desconectado", "nuevomsg": true }, ... ]

---

#### Acción: solicitar perfil de usuario

**URI:** server/usuario/<str:user\_id>

**Metodo:** GET

**Params:** r\_user=<str:appuser\_id>&token=<str:login\_token>

#### Reply

En caso de r\_user, user\_id o token inválido:

- **Response code:** 401 Unauthorized request

En caso de token válido:

Usuario inválido:

- **Response code:** 401 Unauthorized request

Usuario válido:

- **Response code:** 200 OK

---

#### Acción: crear usuario

**URI:** server/usuario/<str:user\_id>

**Metodo:** POST

**Body:** { "password": <str:password>,  
"nombre": <str:nombre>,  
"foto": <str:foto>,  
"ubicación": <str:ubicacion>  
}

*Formato `ubicacion`: string que identifica la ubicación. Ej: FIUBA, Casa, etc*

*Formato `foto`: base64-JPG*

*Los campos `password` y `nombre` son obligatorios.*

#### Reply

Usuario inválido o existente:

- **Response code:** 400 Bad request
- **Body:** { "error": <str:descripcion\_error> }

Usuario válido y atributos inválidos:

- **Response code:** 400 Bad request
- **Body:** { "error": <str:descripcion\_error> }

Usuario válido y atributos válidos:

- **Response code:** 201 Created

---

#### Acción: modificar usuario

**URI:** server/usuario/<str:user\_id>

**Metodo:** PUT

**Params:** r\_user=<str:appuser\_id>&token=<str:login\_token>

**Body:** { "nombre": <str:nombre>,  
"password": <str:password>,  
"foto": <str:foto>,  
"ubicación": <str:ubicacion>  
}

*Nota: Todos los keys son opcionales. Los que no se incluyen retienen su valor original.*

## Reply

En caso de r\_user/token inválido:

- **Response code:** 401 Unauthorized request

En caso de token válido:

Usuario inválido:

- **Response code:** 401 Unauthorized request

Usuario válido y atributos inválidos:

- **Response code:** 400 Bad request
- **Body:** { "error" : <str:descripcion\_error> }

Usuario válido y atributos válidos:

- **Response code:** 201 Created

---

## Recurso - Conversación

**Acción:** solicitar conversación anterior entre usuario <r\_user> y <t\_user>

**URI:** server/conversacion/<str:t\_user>

*Nota: el orden de los argumentos es irrelevante, ambas URI se resuelven por el mismo controlador.*

*Nota2: la conversación se marca como leída en este punto*

**Método:** GET

**Params:** r\_user=<str:appuser\_id>&token=<str:login\_token>

## Reply

*Nota: los mensajes se devuelven en orden ascendente según la fecha y hora de su recepción.*

En caso de r\_user/token inválido:

- **Response code:** 401 Unauthorized request

En caso de token válido:

Usuario(s) inválidos:

- **Response code:** 401 Unauthorized request

Usuario válido y atributos inválidos:

- **Response code:** 400 Bad request
- **Body:** { "error" : <str:descripcion\_error> }

Usuario válido y atributos válidos:

- **Response code:** 200 OK
- **Body:** [ <str:msg1>, <str:msg2>, ... ]

*Format de msg: { "id": <str:user\_id>, "time": <int:timestamp>, "msg": "<str:mensaje>" }*  
*Format de <timestamp>: Unix time*

---

**Acción:** enviar mensaje de usuario <r\_user> a <t\_user>

**URI:** server/conversacion/<str:t\_user>

**Metodo:** POST

**Params:** r\_user=<str:r\_user\_id>&token=<str:login\_token>

**Body:** { "mensaje" : <str:mensaje> }

#### Reply

En caso de r\_user/token inválido:

- **Response code:** 401 Unauthorized request

En caso de token válido:

Usuario(s) o mensaje inválido:

- **Response code:** 400 Bad request
- **Body:** { "error" : <str:descripcion\_error> }

Usuario válido y mensaje válidos:

- **Response code:** 201 Created
- 

**Acción:** enviar mensaje de usuario <r\_user> a todos los usuarios

**URI:** server/broadcast

**Metodo:** POST

**Params:** r\_user=<str:r\_user\_id>&token=<str:login\_token>

**Body:** { "mensaje" : <str:mensaje> }

#### Reply

En caso de r\_user/token inválido:

- **Response code:** 401 Unauthorized request

En caso de token válido:

Mensaje inválido:

- **Response code:** 400 Bad request
- **Body:** { "error" : <str:descripcion\_error> }

Usuario válido y mensaje válidos:

- **Response code:** 201 Created
-