**MÔ TẢ CÁC API ĐỂ LÀM VIỆC VỚI HỆ THỐNG USER CỦA VIOLET**

**/user/login**

Tham số

* username
* password
* src: Tên ứng dụng gọi dịch vụ
* token: mã md5 của xâu <username>violet

Nhiệm vụ:

* Cho biết username và password có hợp lệ không
* Tạo một session trong DB và lưu trữ các thông tin cần thiết trong đó, để việc login được thực hiện trên toàn bộ hệ thống các trang web \*.violet.vn
* Trả về đầy đủ các thông tin người dùng

Mã lỗi errCode có thể trả về:

* errInvalidRequest
* errInvalidRequestData
* errUserNotFound
* errIncorrectPassword
* errUnactivatedUser
* errSystemLoginError (nội dung cụ thể trả về trong errMsg)

**/user/logout**

Tham số: Không có

Nhiệm vụ: Logout khỏi hệ thống (chỉ trên session hiện hành, các sessions khác không bị ảnh hưởng).

**/user/getinfo**

Tham số:

* src: Tên ứng dụng
* Một trong các tham số sau
  + us\_id
  + username
  + email
  + phone

Nhiệm vụ:

* Trả về đầy đủ các thông tin người dùng

Mã lỗi errCode có thể trả về:

* errInvalidRequest
* errUserInvalid (bị chết session)
* errUserNotFound

**/user/update**

Tham số:

* us\_id (có hoặc không): nếu có thì là cập nhật, không có thì là đăng ký mới
* username
* password
* fullname
* gender
* email
* phone
* school
* province
* src: Tên ứng dụng
* token: mã md5 của <us\_id>violet (nếu là cập nhật) hoặc <us\_username>violet (nếu là tạo mới)
* sendmail: là “true” thì sẽ gửi email kích hoạt tài khoản cho người dùng
* oldpass: mật khẩu cũ, dùng trong trường thay đổi mật khẩu hoặc tên truy nhập

Nhiệm vụ:

* Cho phép đăng ký user mới hoặc cập nhật thông tin các user cũ
* Lưu ý nếu là cập nhật thì các tham số thông tin người dùng đều là không bắt buộc

Mã lỗi errCode có thể trả về:

* errInvalidRequest
* errInvalidRequestData (trường hợp đăng ký mới)
* errUserInvalid (trường hợp cập nhật, bị chết session)
* errUserNotFound (trường hợp cập nhật)
* errDataValidation (các dữ liệu không hợp lệ hoặc có từ ngữ bậy bạ, nội dung cụ thể trả về trong errMsg)
* errUsernameExisted
* errEmailExisted
* errPhoneNumberExisted
* errIncorrectOldPassword (trường hợp đổi tên truy nhập hoặc mật khẩu)

**/user/forgotpassword**

Tham số: src, token, email

Nhiệm vụ: Gửi email Quên mật khẩu cho người dùng

Mã lỗi errCode có thể trả về:

* errInvalidRequest
* errEmailNotFound

**/user/cardcharge**

Nhiệm vụ: Nạp tiền vào tài khoản bằng thẻ điện thoại

Tham số:

* us\_id
* type: VIETTEL, MOBIFONE, VINAPHONE
* pin: mã thẻ cào
* serial: mã serial của thẻ cào
* src: tên ứng dụng
* token: mã md5 của <us\_id>violet

Mã lỗi errCode có thể trả về:

* errInvalidRequest
* errUserNotFound
* errCardChargeX (X là mã lỗi, nội dung cụ thể trả về trong errMsg)

**/user/getcurrentuser**

Tham số: Không có

Nhiệm vụ: Trả về user\_id và các thông tin cơ bản của người dùng trên session hiện hành.

**/user/getonlineuser**

Nhiệm vụ: Trả về danh sách những người dùng đang online

Tham số:

* src: Tên ứng dụng
* max: Số lượng tối đa trong danh sách trả về

**/user/noop**

Tham số: Không có gì

Nhiệm vụ: Không làm gì

Lưu ý: Với các ứng dụng offline, flash hoặc JS thì nên gửi NOOP khoảng 2 phút 1 lần để giữ session.

**Lưu ý chung:**

Nếu các hàm API trên được gọi từ các ứng dụng trên server (không phải từ JS hoặc Flash), thì tất cả đều phải thêm một tham số sid là session\_id của trình duyệt gọi ứng dụng này.

Ví dụ:

function RequestWithSession($url, $session) {

$ch = curl\_init($url);

curl\_setopt($ch, CURLOPT\_POST, true);

curl\_setopt($ch, CURLOPT\_POSTFIELDS, 'sid='.$session);

curl\_setopt($ch, CURLOPT\_RETURNTRANSFER, true);

$response = curl\_exec($ch);

curl\_close($ch);

parse\_str($response, $data);

return $data;

}

ini\_set('session.cookie\_domain', 'violet.vn');

session\_name('symfony');

session\_start();

$ss\_user = RequestWithSession('http://api.violet.vn/user/getcurrentuser', session\_id());

**Nội dung các mã lỗi (các ứng dụng tự define)**

errInvalidRequestData=Không đủ dữ liệu đầu vào

errUserNotFound=Tài khoản không tồn tại

errIncorrectPassword=Mật khẩu không chính xác

errUnactivatedUser=Tài khoản chưa xác nhận thông tin liên hệ

errUserInvalid=Tình trạng đăng nhập đã bị ngắt. Hãy mở menu "Tài khoản" và chọn "Đăng nhập lại"

errUsernameExisted=Tên truy nhập bị trùng với 1 tài khoản khác

errEmailExisted=Email bị trùng với 1 tài khoản khác

errPhoneNumberExisted=Số điện thoại bị trùng với 1 tài khoản khác

errIncorrectOldPassword=Mật khẩu cũ không có hoặc không chính xác

errEmailNotFound=Email chưa được đăng ký cho tài khoản nào

errMoneyNotEnough=Không đủ tiền trong tài khoản, hãy nạp thêm tiền vào tài khoản bằng thẻ cào điện thoại hoặc tin nhắn SMS

**SPACE API**

**/space/dir**: Liệt kê danh sách thư mục con và các file trong một thư mục, kể cả thư mục parent .. Riêng với các file có thể filter theo extension.

* **id** (optional): id của thư mục, nếu không có thì là thư mục gốc
* **filter** (optional): các extension cách nhau dấu chấm phẩy, ví dụ “doc;ppt;xls”
* Trả về danh sách file

<

cần thêm tham sô:

* dironly (optional): =1 🡺 chỉ lấy ra danh sách thư mục.

Trong record của thư mục cần thêm trường chỉ ra số lượng child (dir và file)của thư mục đó để không phải đọc db nhiều lần.

Đối với các file ảnh cần trường lưu thông tin về thumbnail, với file Word, Excel, PDF cần thông tin ảnh preview (xem code API của phần space version cũ)

>

**/space/mkdir**: Tạo 1 thư mục mới

* **parent\_id** (optional): id của thư mục cha, nếu không có thì là thư mục gốc
* **name**: Tên của thư mục mới
* Trả về thông tin thư mục mới

**/space/delete**: Xóa một hoặc nhiều file hoặc thư mục

* **id**: có thể là 1 id hoặc nhiều id của file hoặc thư mục, cách nhau bằng dấu phẩy
* **option**: force

<

Cần trả về code kiểm tra số record affected hoặc danh sách các id bị xóa để đảm bảo tính chính xác khi xóa phía client

>

**/space/upload**: Upload một file hoặc một gói file (thư mục nén thành file zip)

* **dir** (optional): id của thư mục, không có thì là thư mục gốc
* **unpack** (optional): Nếu bằng true thì sau khi upload sẽ bung gói thành nhiều file và thư mục
* Trả về danh sách các file đã được đưa lên

<

Khi upload hoặc save file thì cần tạo thumbnail (đối với file ảnh) và ảnh preview ( với file Word, Excel, PDF)

>

**/space/save**: Lưu một bộ dữ liệu (trên memory) trực tiếp vào file trên Space

1. **fileid** (optional): id của file sẽ được lưu vào, nếu không có là tạo file mới
2. Nếu không có fileid, cần các tham số sau:
   * **dir** (optional): id của thư mục, nếu không có thì là thư mục gốc
   * **name**: Tên của file sẽ lưu vào
3. Trả về thông tin của file

**/space/copy**: Copy một hoặc nhiều file/thư mục ra một thư mục khác

* **id**: có thể là 1 id hoặc nhiều id của file và thư mục, cách nhau bằng dấu phẩy
* **destination** (optional): id của thư mục đích, nếu không có thì là thư mục gốc
* Trả về danh sách các file đã được copy

**/space/move**: Chuyển một hoặc nhiều file/thư mục ra một thư mục khác, vẫn giữ nguyên id

* **id**: có thể là 1 id hoặc nhiều id của file và thư mục, cách nhau bằng dấu phẩy
* **destination** (optional): id của thư mục đích, nếu không có thì là thư mục gốc

**/space/rename**: Đổi tên file hoặc thư mục

* **id**: id của file của thư mục
* **name**: Tên mới của file hoặc thư mục đó

**/space/file**

* **id**: id của file cần lấy nội dung
* **relative** (optional), đường dẫn tương đối tới các file phụ thuộc
* **filename** (optional)
* Trả về nội dung file

Trong trường hợp có lỗi sẽ trả về các mã lỗi và message tương ứng trong một cấu trúc XML

< status errCode=”…” err=”…”>