Hướng dẫn tích hợp GAMIFICATION PLATFORM

**Version <1.0.0>**

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Ngày** | **Phiên bản** | **Mô tả** | **Người thực hiện** |
| 27/06/2022 | 1.0.0 |  | NamBV |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Mục Lục**

[**Hướng dẫn tích hợp minigame Vòng Quay May Mắn**: 5](#_Toc107319610)

[1: Chơi game trực tiếp trên trang landing page 5](#_Toc107319611)

[2: Nhúng game vào nội dung trang web. 7](#_Toc107319612)

[3: Chơi game trên popup của trang đích 9](#_Toc107319613)

[**Tạo game theo yêu cầu tùy chỉnh** 11](#_Toc107319614)

[Bước 1: Tạo event game trên backend 11](#_Toc107319615)

[Bước 2: Tạo nội dung game 11](#_Toc107319616)

[Bước 3: Nhúng SDK Game vào web 11](#_Toc107319617)

[Bước 4: Nhúng mã khởi tạo SDK 12](#_Toc107319618)

[Bước 5: Các Hàm tính năng cho game 12](#_Toc107319619)

**Hướng dẫn tích hợp minigame Vòng Quay May Mắn**:

Game hiện tại đang có 3 hướng tích hợp,

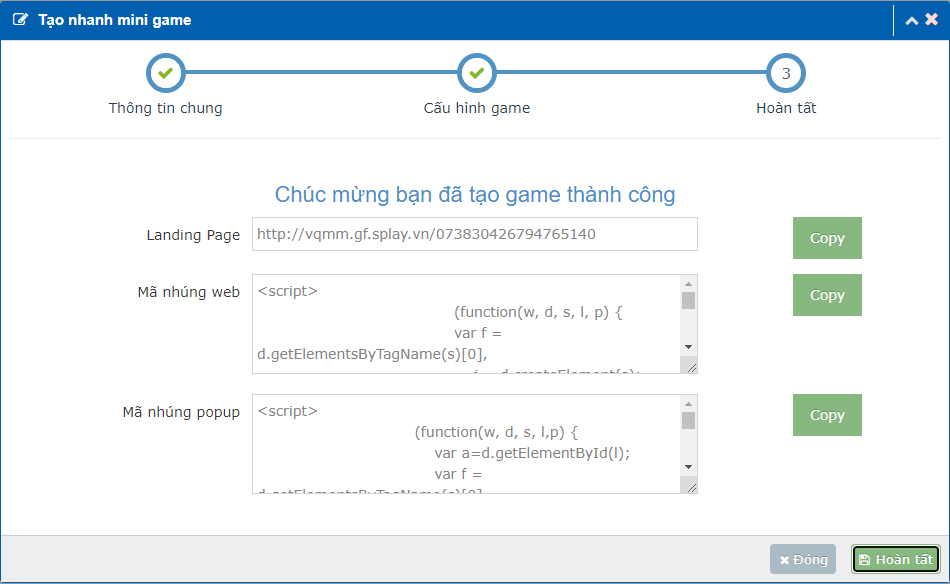
1: Chơi game trực tiếp trên trang landing page

<http://vqmm.gf.splay.vn/code>

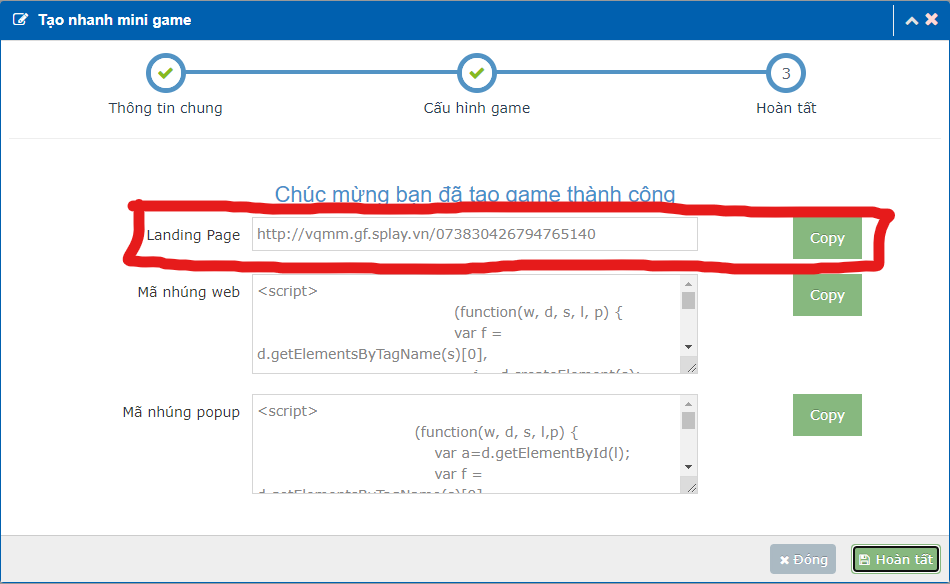
* **Mục đích:**

Cho phép dự án tích hợp minigame Vòng Quay may mắn vào ứng dụng, dịch vụ của mình. User tham gia chơi có thể chơi game trực tiếp trên landing page

* **Cách tích hợp:**
  + B1: Truy cập vào <http://backend.gf.splay.vn/> đăng nhập với bằng tài khoản mail VTC > Vào menu game trong > Vào Danh sách game > Click nút “Tạo nhanh minigame” để hiển thị popup Tạo game > Click nút “Tiếp theo” vào tab Cấu hình game



* + B2: Copy địa chỉ landing page để lấy link game và phát hành tùy theo mục đích.

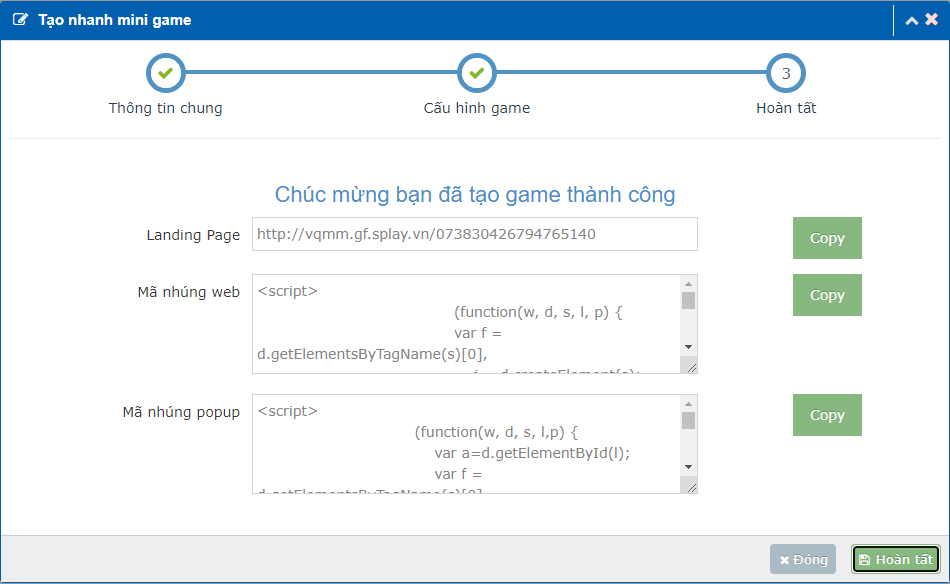


2: Nhúng game vào nội dung trang web.

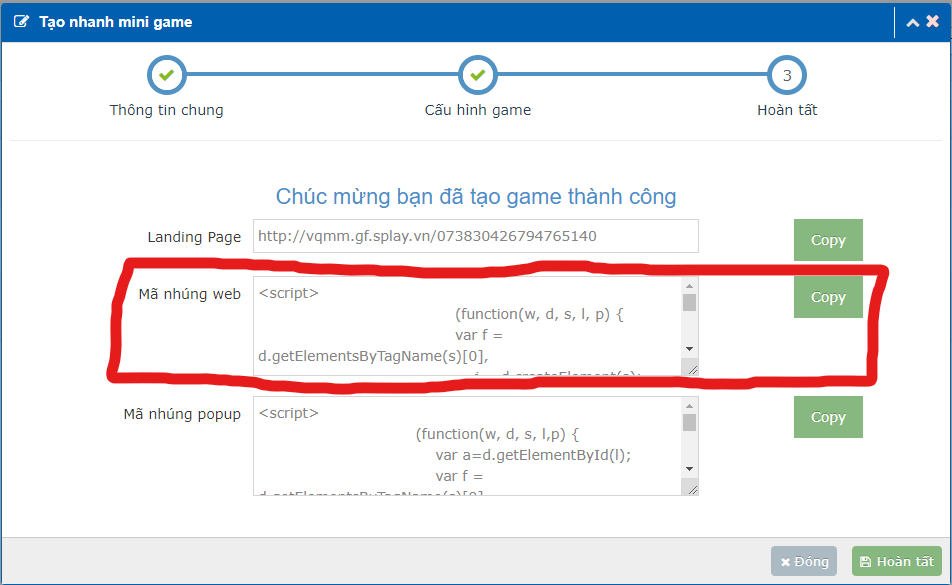
* **Mục đích:**

Cho phép dự án tích hợp minigame Vòng Quay may mắn vào ứng dụng, dịch vụ của mình. Nhúng game vào web hoặc app ở vị trí mong muốn

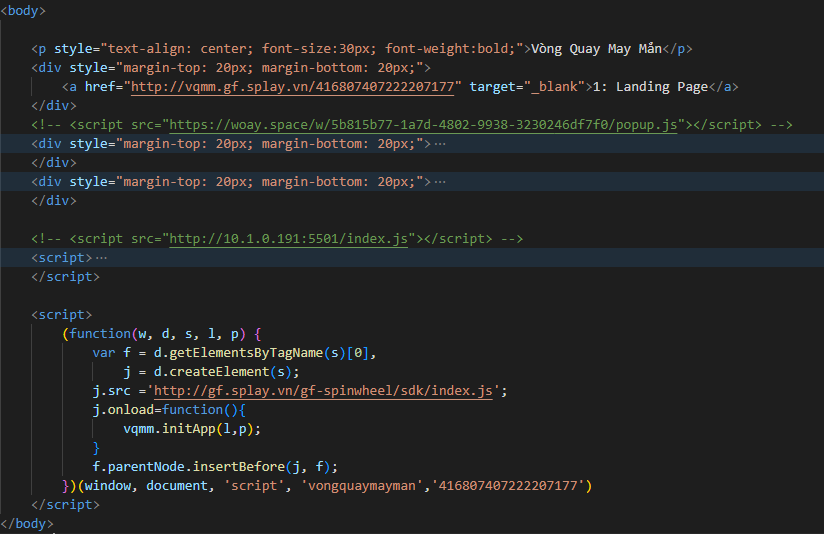
* **Cách tích hợp:**
  + B1: Truy cập vào <http://backend.gf.splay.vn/> đăng nhập với bằng tài khoản mail VTC > Vào menu game trong > Vào Danh sách game > Click nút “Tạo nhanh minigame” để hiển thị popup Tạo game > Click nút “Tiếp theo” vào tab Cấu hình game



* B2: Copy đoạn mã js được khoang vùng đỏ bên dưới:



* B3: Sau đó dán vào bên trang html trong thẻ body ở 1 vị trí bất kì mà bạn muốn nó sẽ hiển thị:

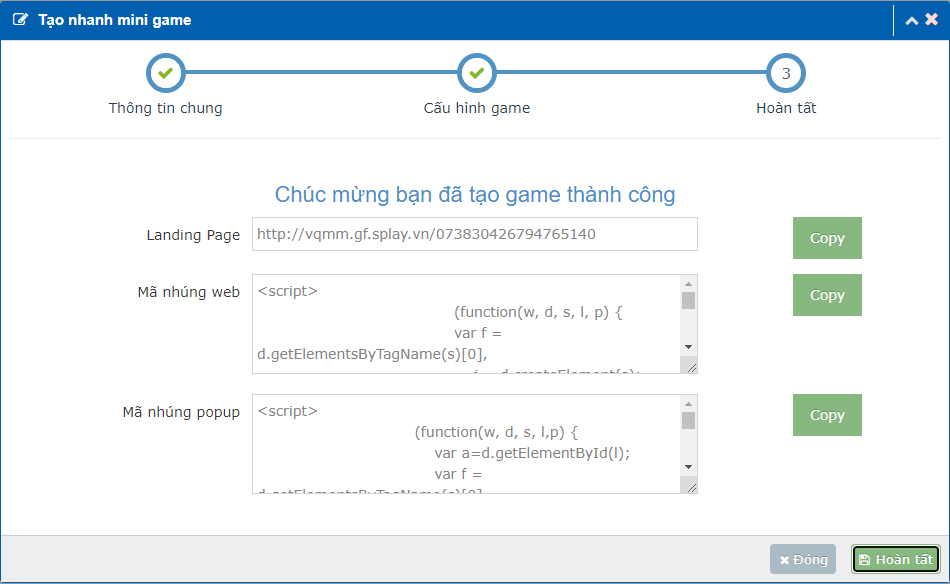


3: Chơi game trên popup của trang đích

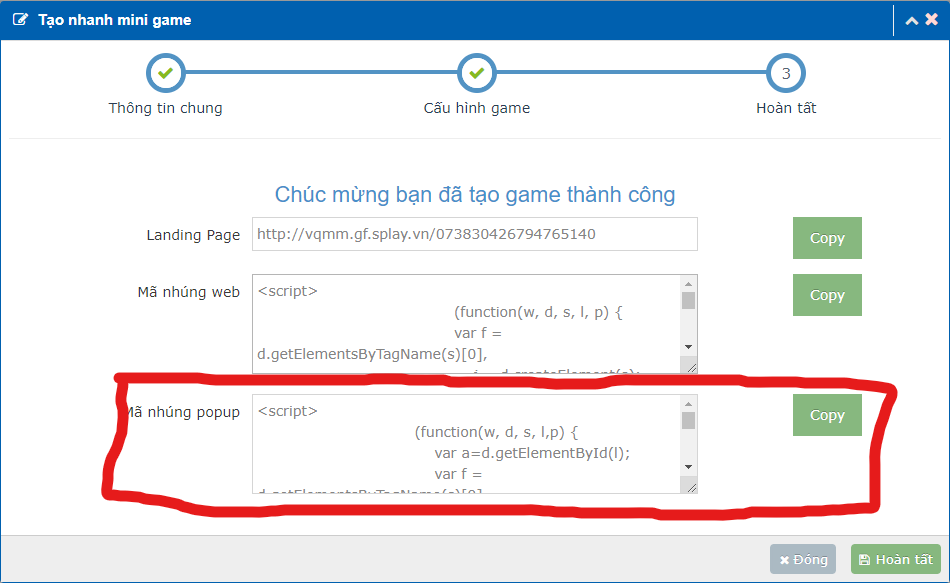
* **Mục đích:**

Cho phép dự án tích hợp minigame Vòng Quay may mắn vào ứng dụng, dịch vụ của mình. Có thể chơi game trên popup của web đang chạy

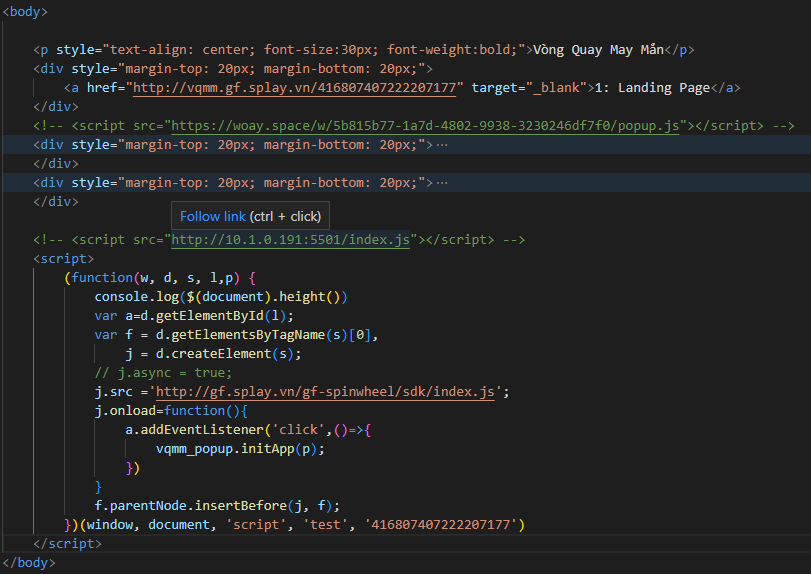
* **Cách tích hợp:**
  + B1: Truy cập vào <http://backend.gf.splay.vn/> đăng nhập với bằng tài khoản mail VTC > Vào menu game trong > Vào Danh sách game > Click nút “Tạo nhanh minigame” để hiển thị popup Tạo game > Click nút “Tiếp theo” vào tab Cấu hình game



* B2: Copy đoạn mã js được khoang vùng đỏ bên dưới:



* B3: Sau đó dán vào bên trang html trên thẻ đóng </ body>:



Tạo game theo yêu cầu tùy chỉnh

Để triển khai minigame trên giao diện web theo yêu cầu tùy chỉnh thì cần có SDK Game. Để tích hợp thực hiện theo các bước sau:

Bước 1: Tạo event game trên backend

Truy cập vào <http://backend.gf.splay.vn/> đăng nhập với bằng tài khoản mail VTC > Vào menu game trong > Vào Danh sách game > Click nút “Thêm mới” để hiển thị popup Tạo game > Nhập thông tin cho game

Bước 2: Tạo nội dung game, vật phẩm, tỷ lệ…

Tab menu bên trái tạo ‘Chế độ game’, ‘Phòng game’, click vào ‘Kho phần thưởng’ để tạo phần thưởng, vật phẩm

Bước 3: : Nhúng SDK Game vào website:

Copy đoạn mã bên dưới vào trước thẻ đóng </body>

 <script src="http://static.gf.splay.vn/gf-sdk/v1.0/vendor/axios.min.js?v=1"></script>

  <script src="http://static.gf.splay.vn/gf-sdk/v1.0/vendor/jquery.min.js?v=1"></script>

  <script src="http://static.gf.splay.vn/gf-sdk/v1.0/vendor/moment-with-locales.min.js?v=1"></script>

  <script src="http://static.gf.splay.vn/gf-sdk/v1.0/app.js?v=1"></script>

  <script src="http://static.gf.splay.vn/gf-sdk/v1.0/auth.js?v=1"></script>

  <script src="http://static.gf.splay.vn/gf-sdk/v1.0/common.js?v=1"></script>

  <script src="http://static.gf.splay.vn/gf-sdk/v1.0/event.js?v=1"></script>

Bước 4: Nhúng mã khởi tạo

- **initApp**: Hàm khởi tạo event, kiểm tra người chơi đã login hay chưa, cài đặt config cho sdk,

biến truyền vào là object config

 const vtcmAppConfig = {

        gameId:100004,

        apiBaseUrl: 'http://api.gf.splay.vn',

        client\_id: "GF\_EVENTS\_WEB",

        client\_secret: "aP6k2Ql68arPH8l",

        url\_return:`http://tramtien.splay.vn/mungdoclap`,

        path:'/mungdoclap'

};

gameId: mã được tạo ra lúc tạo game trên trang admin game

apiBaseUrl: api server có thể truyền hoặc để trống

client\_id: id người dùng, được cấp cho từng trung tâm

client\_secret: mã bí danh của người dùng, được cấp cho từng trung tâm

url\_return: url của trang sự kiện

path: path sau domain

Bước 5: Các Hàm tính năng cho game

- **getAppSetting**: Hàm lấy về dữ liệu cài đặt giao diện cho sự kiện,

biến truyền vào là 1 callback function

- **rollup**: Hàm điểm danh nhận thưởng cho người chơi,

biến truyền vào gồm:

modeId: id chế độ chơi game,

roomId: id phòng game

handlingRollup: Callback function xử lý dữ liệu trả về, hàm truyền vào 2 biến là:

roomId, response(dữ liệu trả về từ server)

errRollup: Xử lý lỗi xảy ra, hàm truyền vào 2 biến là:

roomId, error(lỗi server trả về)

- **playGame**: Hàm trả về kết quả của các lượt chơi,

biến truyền vào gồm:

modeId: id chế độ chơi game,

numberPlayed: số lượt chơi (vd: 1 hay 10)

objectParamReturn: object chứa danh sách biến cần cho xử lý giao diện

handlingPlayGame: Callback function xử lý dữ liệu trả về, hàm truyền vào 2 biến là:

objectParamReturn: danh sách các biến,

response : dữ liệu trả về từ server

errPlayGame: Xử lý lỗi xảy ra, hàm truyền vào biến error(lỗi server trả về)

setStatusGame: Callback function trả về trạng thái cho game khi api bị lỗi

- **getHistory**: Hàm lấy về lịch sử chơi,

biến truyền vào gồm:

page: vị trí trang cần lấy,

modeId: chế độ chơi của game,

rewardType:

handlingGetHistory: Callback function xử lý dữ liệu trả về, hàm truyền vào 4 biến là:

page, modeId, rewardType, response(dữ liệu trả về từ server)

errHistory: Xử lý lỗi xảy ra, truyền vào biến error(lỗi server trả về)

- **exchangeRewards**: Hàm đổi thưởng cho người chơi,

biến truyền vào gồm:

modeId: chế độ chơi của game,

value: mốc đổi thưởng,

handlingExchangeRewards: Callback function xử lý dữ liệu trả về, hàm truyền vào là

modeId, value, respone(dữ liệu trả về từ server)

errExchangeRewards: Xử lý lỗi xảy ra, truyền vào biến error(lỗi server trả về)

- **exchangeRewardsWithMilestones**: Hàm đổi thưởng theo thứ hạng người chơi,

biến truyền vào gồm:

modeId: chế độ chơi của game,

value: mốc đổi thưởng,

handlingExchangeRewardsWithMilestones: Callback function xử lý dữ liệu trả về, hàm truyền vào là

modeId, value, respone(dữ liệu trả về từ server)

errExchangeRewardsWithMilestones: Xử lý lỗi xảy ra, truyền vào biến error (lỗi server trả về)

- **getBXHPayment**: Hàm lấy về bảng xếp hạng người chơi,

Biến truyền vào gồm:

handlingGetBXHPayment: Callback function xử lý dữ liệu trả về, hàm truyền vào là

respone(dữ liệu trả về từ server)

errBXH: Xử lý lỗi xảy ra, truyền vào biến error(lỗi server trả về)

-**getInviteCodeForShare**: Hàm lấy về mã code dùng cho việc người dùng chia sẻ link web cho bạn bè

Biến truyền vào gồm:

modeId: id chế độ chơi game,

roomId: id phòng chơi,

objectParamReturn: object chứa danh sách biến cần cho xử lý giao diện

handlingGetInviteCodeForShare: Callback function xử lý dữ liệu trả về, hàm truyền vào 2 biến là:

objectParamReturn: danh sách các biến,

response : dữ liệu trả về từ server

errGetInviteCode: Xử lý lỗi xảy ra, hàm truyền vào biến error(lỗi server trả về)

-**sendInviteCode**: Gửi mã code lên cho server xác nhận

Biến truyền vào gồm:

modeId: id chế độ chơi game,

roomId: id phòng chơi,

code: mã code

handlingSendInviteCode: Callback function xử lý dữ liệu trả về, hàm truyền vào biến là:

response : dữ liệu trả về từ server

errSendInviteCode: Xử lý lỗi xảy ra, hàm truyền vào biến error(lỗi server trả về)

-**getUserData**: Lấy thông tin chi tiết người chơi

Biến truyền vào gồm:

listParams: danh sách các biến truyền lên server,

handlingGetUserData: Callback function xử lý dữ liệu trả về, hàm truyền vào biến là:

response : dữ liệu trả về từ server

errGetUserData: Xử lý lỗi xảy ra, hàm truyền vào biến error(lỗi server trả về)

-**setDataUserInGame**: Chọn thông tin nhân vật trong game của người chơi

Biến truyền vào gồm:

listParams: danh sách các biến truyền lên server,

objectParamReturn: danh sách các biến,

handlingUpdateDataUserInGame: Callback function xử lý dữ liệu trả về, hàm truyền vào 2 biến là:

response : dữ liệu trả về từ server

objectParamReturn: danh sách các biến,

errSetDataUserInGame: Xử lý lỗi xảy ra, hàm truyền vào biến error(lỗi server trả về)

-**updateDataUser**: Cập nhật thông tin chi tiết của người chơi

Biến truyền vào gồm:

listParams: danh sách các biến truyền lên server,

objectParamReturn: danh sách các biến,

handlingUpdateDataUser: Callback function xử lý dữ liệu trả về, hàm truyền vào 2 biến là:

response : dữ liệu trả về từ server

objectParamReturn: danh sách các biến,

errUpdateDataUser: Xử lý lỗi xảy ra, hàm truyền vào biến error(lỗi server trả về)

-**getLocation**: Lấy danh sách thông tin tỉnh, huyện, xã

Biến truyền vào gồm:

listParams: danh sách các biến truyền lên server,

objectParamReturn: danh sách các biến,

handlingGetLocation: Callback function xử lý dữ liệu trả về, hàm truyền vào 2 biến là:

response : dữ liệu trả về từ server,

objectParamReturn: danh sách các biến,

errGetLocation: Xử lý lỗi xảy ra, hàm truyền vào biến error(lỗi server trả về)

-**getLocation**: Lấy danh sách thông tin tỉnh, huyện, xã

Biến truyền vào gồm:

listParams: danh sách các biến truyền lên server,

objectParamReturn: danh sách các biến,

handlingGetLocation: Callback function xử lý dữ liệu trả về, hàm truyền vào 2 biến là:

response : dữ liệu trả về từ server,

objectParamReturn: danh sách các biến,

errGetLocation: Xử lý lỗi xảy ra, hàm truyền vào biến error(lỗi server trả về)