

















Data Plan Details






Server Routes

Server Routes

 Method	 URL	 Purpose	 Request	 Response
	Public Access:			
<u>Post</u>	/auth/login	Authorize User Login	password, username or email address	a JWT token and user details object
<u>Post</u>	/auth/logout	User Logout	JWT token	success code
<u>Post</u>	/auth/register	User Registration (venue management and admin need further approval)	password, username, email address, phone number, user role	success code:20000
<u>GET</u>	/public/hotspot	Retrieve hotspot location	JWT token	A array of current hotspot location then render location to map
	Authorized Access:			
<u>GET</u>	/user/	Retrieve All User Info, admin only	JWT token	All user details objects in an array

 Method	 URL	 Purpose	 Request	 Response
<u>GET</u>	/user/info/{userid}	retrieve user details by id	user id, and JWT token	single user details object, and success code
<u>Post</u>	/user/info/{userid}	Update User Info	Post a modified existing user details object which retrieved from get user by id	success code and processed user details object
<u>DELETE</u>	/user/info/{userid}	Delete User (admin only)	user id, and JWT token	success code
<u>GET</u>	/user/role/{level}	Retrieve All User in specific level. For example, admin can get all level user info, venue manager only allowed to get guest info	JWT token and Level in number	Specific Level user details objects in an array
<u>Post</u>	/user/checkIn/{userId}	User check into the venue	user id, location, timestamp, venue id, and JWT token	success code
<u>GET</u>	/user/checkIn/{userId}	Retrieve User check in history	user id, and JWT token	Array contains user checkin history and been in hotpot or not

 Method	 URL	 Purpose	 Request	 Response
<u>GET</u>	/user/message/{userId}	Retrieve User message	user id, and JWT token	Array contains user message
	Venue Manager Access:			
<u>GET</u>	/venue/venueID/{venueid}	Retrieve venue information	venue id, JWT token	Single Venue information in a object.
<u>Post</u>	/venue/venueID/{venueid}	Modify venue information	Post a modified existing venue details object which retrieved from get venue by id	success code and processed information object
<u>POST</u>	/venue/qrcode/{venueid}	Request new qrcode	venue id, JWT token	a new url contains qrcode
<u>GET</u>	/venue/qrcode/{venueid}	Retrieve qrcode	venue id, JWT token	a new url contains qrcode
<u>GET</u>	/venue/history/{venueid}	Retrieve user details which has been visited venue	venue id, JWT token	An Array of User details, and user checkin history
	Admin/ Medical Official Access:			
<u>GET</u>	/admin/approval/	GET approvals pending list	JWT token	A array of user details which is pending approvals

 Method	 URL	 Purpose	 Request	 Response
<u>PATH</u>	/admin/approval/{userid}	Approve venue manager and new admin into system	user id, and JWT token	Success Code
<u>POST</u>	/admin/hotspot	Make a flag for new hotspot	Location, time, JWT token	Success code, and A array of current hotspot location
<u>GET</u>	/admin/hotspot	Retrieve hotspot location	JWT token	A array of current hotspot location then render location to map

Object Example TODO

User Info response json example :

```
{
  "update_time": "2021-05-09T13:36:45.000Z",
  "create_time": "2021-05-09T13:36:47.000Z",
  "update_by": "Akide_Liu",
  "create_by": "Akide_Liu",
  "user_id": 1,
  "username": "Akide_Liu",
  "password": "123456",
  "address": "5 allan st, prospect, SA",
  "emailAddress": "mail@llycloud.com",
  "is_in_hotspot": false,
  "is_approval": true,
  "phone": 450460166,
  "Role": [
    {
      "update_time": "2021-05-09T13:37:33.000Z",
      "create_time": "2021-05-09T13:37:35.000Z",
      "update_by": "Akide_Liu",
      "create_by": "Akide_Liu",
      "role_id": 1,
    }
  ]
}
```



```



        "level": 3,
        "role_name": "admin",
        "description": "administrators"
    }
],
"UserCheckIn": [
    {
        "update_time": "2021-05-09T14:16:06.000Z",
        "create_time": "2021-05-09T14:16:07.000Z",
        "update_by": "Akide_Liu",
        "create_by": "Akide_Liu",
        "check_in_id": 1,
        "venue_id": 1,
        "CheckInAddress": "156 george st ,Adelaide",
        "CheckInTimeStamp": "2021-05-09T14:16:30.000Z"
    }
],
"Venue": [
    {
        "update_time": "2021-05-09T14:23:38.000Z",
        "create_time": "2021-05-09T14:23:39.000Z",
        "update_by": "Akide_Liu",
        "create_by": "Akide_Liu",
        "venue_id": 1,
        "check_in_code": "1000001",
        "email_address": "mail@adelaide.edu.au",
        "venue_name": "University of Adelaide",
        "shop_address": "The University of Adelaide\nSA 5005\nAUSTRALIA",
        "is_hotspot": false
    }
]
}

```

Client/Server Interactions

Client/Server Interactions

 Page	 Interaction
<u>All</u>	Each page is served up statically by the server
<u>Home</u>	The Client request hotspot and render to map, different level of user might have different view. higher level authority have some manage features
<u>Personal Information</u>	The Client request specific user info it should display and can be modify.
<u>User Information</u>	The Admin request all user info and render as a table it should display and can be modify.

 Page	 Interaction
<u>venue information</u>	The venue manager request belonged venue and update venue information
<u>Admin</u>	The Admin or Medical official request admin interface it should display all information in entire system and administrative features.