Services API

* Base URL: <http://45.55.59.22:8080/GamificationMaven>
* For un-authenticated requests return [un-authenticated response](#Unauthenticated_Response_Obj)

|  |  |  |  |
| --- | --- | --- | --- |
| Service Uri | Description | Input | Output |
| /login  [POST] | Authenticate users through their email & base64 password  N.B: [Invalid Login](#Invalid_Pass_Login_Obj) | {  email: String,  password: Base64  } | [Response](#Response_Obj) with  data:{  user: [user\_min](#User_Min_Obj),  token: [token](#Token_Obj)  }  N.B: password should be excluded |
| /register  [POST] | Store new user in DB with **role:”User”** | [user](#User_Obj)  N.B: coins & experience will be sent as 0 value.  //ToDO: activation mail should be sent to registered user to update active\_account field | [Response](#Response_Obj) with  data:{  user: [user\_min](#User_Min_Obj)  }  N.B: password should be excluded |
| /users  [POST] | Retrieve all **active** registered users **with role User**  N.B: if token expired [invalid\_token](#invalid_token_obj) obj will be sent | token is sent in Authorization header, then we verify it on server through JWT | [Response](#Response_Obj) with  data:{  users: [{[user\_min](#User_Min_Obj)}]  }  N.B: users is array of user minified obj |
| /getprofile | Get User profile by id | token Authorization Header | [Response](#Response_Obj) with  data:{  users: [{[user](#User_Obj)}]  } |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Objects

|  |  |
| --- | --- |
| Object | Description |
| response | {  errorCode: Int, // 0: No Error 1: Error Occurred  errorMessage: String,  data: Obj  } |
| token | {  token: String,  expires: String //Sec  } |
| invalid\_token | {  errorCode: 2,  errorMessage: "Invalid Token",  data: null  } |
| un-authenticated | {  errorCode:1,  errorMessage: “unauthenticated request”,  data: null  } |
| user | {  first\_name: String,  last\_name: String,  mail: String,  password: Base64 String,  nick\_name: String [Nullable],  pic:  coins: Int,  experience: Int,  mobile: String [Nullable],  address: String [Nullable],  active\_account: Boolean,  role: String  } |
| user\_min | {  first\_name: String,  last\_name: String,  mail: String,  \_id: Int //Unique,  nick\_name: String  } |
| invalid\_login | {  errorCode:1,  errorMessage:” Invalid Username / Password”,  data:null  } |
|  |  |
|  |  |
|  |  |
|  |  |