

# FollowUP

Follow UP APP

## Users Model

### 1. User ADD

**Link :** <http://followup.heartgate.co/api/user/add> **Method :** POST

**Request :** [name, username, email, role\_id, team\_id, phone, address, bio, skills[], password, password\_confirmation]

**Response :** ['data' => 'User Added successfully', 'state'=>'1']

**Validation :** username & email are unique - all data are required

### 2. Login user

**Link :** <http://followup.heartgate.co/api/login> **Method :** POST

**Request :** [email, password]

**Response :** ['data' => User Data, 'state'=>'1']

**Validation :** user can login with username or email

### 3. Update user image [base64]

**Link :** <http://followup.heartgate.co/api/update-user-image> **Method :** POST

**Request :** [user\_id, image]

**Response** : ['data' => Image path , 'state'=>'1']

**Validation** : accept base64 for image and integer user\_id - all data are required

#### 4. Change Password

**Link** : <http://followup.heartgate.co/api/change-password> **Method** : POST

**Request** : [user\_id, oldPassword, newPassword, newPassword\_confirmation]

**Response** : ["data"=>"Password Updated successfully", "state"=>1]

**Validation** : accept integer values for (user\_id) string for (oldPassword-newPassword)- all data are required min:6 for passwords

#### 5. Update Player Id

**Link** : <http://followup.heartgate.co/api/update-player-id> **Method** : POST

**Request** : [user\_id, player\_id]

**Response** : ["data"=>"Data Updated Successfully", "state"=>1]

**Validation** : accept integer values for (user\_id) and string for (player\_id) - all data are required

#### 6. User Information

**Link** : <http://followup.heartgate.co/api/user-info> **Method** : POST

**Request** : [user\_id]

**Response** : ["data"=> User DATA, "state"=>1]

**Validation** : accept integer values for (user\_id) - all data are required

#### 7. Update user information

**Link** : <http://followup.heartgate.co/api/update-user> **Method** : POST

**Request** : [user\_id, name, phone, address, role\_id, team\_id, bio, skills]

**Response** : ["data"=> User DATA, "state"=>1]

**Validation** : accept integer values for (user\_id- phone)- all data are required

## 8. Reset Password

**Link :** <http://followup.heartgate.co/api/reset-password> **Method :** POST

**Request :** [email]

**Response :** ["data"=> success message, "state"=>1]

**Validation :** email string & Required [must be exist in users table]

## 9. Delete Account

**Link :** <http://followup.heartgate.co/api/delete-user/{user}> **Method :** POST

**Request :** [password]

**Response :** ["data"=> success message, "state"=>1]

# Skills Model

## 10. Get All Skills

Link : <http://followup.heartgate.co/api/skills> Method : GET

Request :

Response : ['data' => All skills, 'state'=>'1']

Validation :

## 11. User with this skill

Link : <http://followup.heartgate.co/api/skill/{skill}/users> Method : GET

Request : [skill\_id]

# Teams Model

## 12. Get All teams

Link : <http://followup.heartgate.co/api/departments> Method : GET

Request :

Response : ['data' => All teams, 'state'=>'1']

Validation :

## 13. User with this team

Link : <http://followup.heartgate.co/api/department/{team}/users> Method : GET

Request : [team\_id]

# Roles Model

## 14. Get All roles

Link : <http://followup.heartgate.co/api/roles> Method : GET

Request :

Response : ['data' => All roles, 'state'=>'1']

Validation :

## 15. User with this Role

Link : <http://followup.heartgate.co/api/role/{role}/users> Method : GET

Request : [role\_id]

# Suppliers Model

## 16. Get All suppliers

Link : <http://followup.heartgate.co/api/suppliers> Method : GET

Request :

Response : ['data' => All roles, 'state'=>'1']

Validation :

## 17. Show one Supplier

Link : <http://followup.heartgate.co/api/supplier/{supplier}> Method : GET

Request : [role\_id]

# Procurement Model

## 18. Create New Procurement [Add new procurement to system]

Link : <http://followup.heartgate.co/api/procurement/create> Method : POST

Request : ['user\_id', 'supplier\_id', 'company\_name', 'po\_number', 'deadline', 'items', 'details', 'place']

Response : ['data' => Procurement Added Successfully, 'state'=>'1']

Validation : date '02/21/2019' => ' mm/dd/YYYY'

## 19. Get user Procurements [all procurements that belong to one user]

Link : <http://followup.heartgate.co/api/procurements/{user}> Method : GET

Request : [user\_id]

## 20. Show one Procurement[show one procurement in detail]

Link : <http://followup.heartgate.co/api/procurement/{procurement}> Method : GET

Request : [user\_id]

## 21. Procurement manager show files [Procurement manager view all procurements]

Link : <http://followup.heartgate.co/api/manager/{user}/procurements> Method : GET

Request : [user\_id]

## 22. Create New Offer Price [offer price that created with procurement manager]

Link : <http://followup.heartgate.co/api/offer-price/{procurement}/create> Method : POST

Request : ['user\_id', 'notes', 'attaches']

Response : ['data' => Show Procurement Data, 'state'=>'1']

### **23. Request Demo [user can request Demo after offer price]**

**Link :** <http://followup.heartgate.co/api/procurement/{procurement}/demo> **Method :** POST

**Request :** ['user\_id']

**Response :** ['data' => Success Message, 'state'=>'1']

### **24. Request Execute [user can request execution after offer price]**

**Link :** <http://followup.heartgate.co/api/procurement/{procurement}/execute> **Method :** POST

**Request :** ['user\_id']

**Response :** ['data' => Success Message, 'state'=>'1']

### **25. Confirm Or Reject Execution [manager of Accountant can accept or reject any demo or execution request]**

**Link :** <http://followup.heartgate.co/api/log/{procurement}/create> **Method :** POST

**Request :** ['user\_id', 'notes', 'attaches', 'status']

**Response :** ['data' => Success Message, 'state'=>'1']

**Validation :** status=> '0 or 1' 0 for reject and 1 for accept

# Ideas Model

## 26. Get Ideas for a user or manager

**Link :** <http://followup.heartgate.co/api/my-ideas/{user}> **Method :** GET

**Request :** ['user\_id']

**Response :** ['data' => user ideas, 'state'=>'1']

## 27. Add New Idea

**Link :** <http://followup.heartgate.co/api/my-ideas> **Method :** POST

**Request :** ['subject', managers, 'user\_id', 'attaches']

**Response :** ['data' => Idea, 'state'=>'1']

**Validation :** int 'user\_id', array 'managers', file 'attaches'

## 28. Add Reply

**Link :** <http://followup.heartgate.co/api/idea-reply/{idea}> **Method :** POST

**Request :** ['idea', 'description', 'user\_id', 'attaches']

**Response :** ['data' => Idea, 'state'=>'1']

**Validation :** int 'idea'

## 29. Delete Idea

**Link :** <http://followup.heartgate.co/api/user/{user}/idea> **Method :** POST

**Request :** ['idea']

**Response :** ['data' => Idea Deleted Successfully, 'state'=>'1']

**Validation :** int 'idea'



### 30. View Idea

**Link :** <http://followup.heartgate.co/api/idea/{idea}> **Method :** POST

**Request :** ['user', idea]

**Response :** ['data' => Idea, 'state'=>'1']

**Validation :** int 'user', 'idea'

## Survey Model

### 31. Get All Surveys

**Link :** <http://followup.heartgate.co/api/surveys> **Method :** GET

**Request :** ['user\_id']

**Response :** ['data' => surveys, 'state'=>'1']

### 32. Survey Info

**Link :** <http://followup.heartgate.co/api/survey/{survey}> **Method :** GET

**Request :** ['survey']

**Response :** ['data' => survey, 'state'=>'1']

### 33. Survey Vote

**Link :** <http://followup.heartgate.co/api/survey/{survey}> **Method :** POST

**Request :** ['survey, user\_id, answer\_id']

**Response :** ['data' => survey, 'state'=>'1']

# Poll Model

## 34. Get All Polls

Link : <http://followup.heartgate.co/api/polls> Method : GET

Request : ['user\_id']

Response : ['data' => polls, 'state'=>'1']

## 35. Poll Info

Link : <http://followup.heartgate.co/api/poll/{poll}> Method : GET

Request : [poll]

Response : ['data' => poll, 'state'=>'1']

## 36. Poll Vote

Link : <http://followup.heartgate.co/api/poll/{poll}> Method : POST

Request : [poll, user\_id, answer\_id]

Response : ['data' => poll, 'state'=>'1']

# User of the month Model

## 37. Get UOM

Link : <http://followup.heartgate.co/api/uom> Method : GET

Request : ['user\_id']

Response : ['data' => uom, 'state'=>'1']

## 38. UOM Info

Link : <http://followup.heartgate.co/api/uom/{uom}> Method : GET

Request : [poll]

Response : ['data' => uom, 'state'=>'1']

## 39. UOM Vote

Link : <http://followup.heartgate.co/api/uom/{uom}> Method : POST

Request : [poll, user\_id, answer\_id]

Response : ['data' => uom, 'state'=>'1']

# Documents Model

## 40. Get All Documents

Link : <http://followup.heartgate.co/api/documents> Method : GET

Response : ['data' => documents, 'state'=>'1']

## 41. Add New Document

Link : <http://followup.heartgate.co/api/documents> Method : POST

Request : [user\_id, title, category [1=>'HR', 2=>'MANAGEMENT', 3=>'Announcement', 4=>'Social', 5=>'Other']]

Response : ['data' => Document, 'state'=>'1']

## 42. Get User Documents

Link : <http://followup.heartgate.co/api/user/{user}/documents> Method : GET

Request : [user]

Response : ['data' => documents, 'state'=>'1']

## 43. Delete User Documents

Link : <http://followup.heartgate.co/api/user/{user}/document> Method : GET

Request : [user, document]

Response : ['data' => deleted successfully, 'state'=>'1']