**Emcure Compliance: Backend APIs**

* PATH: /listusers Method: GET

Input: [http://localhost:8082/listusers?page=1 HYPERLINK "http://localhost:8082/listusers?page=1&per\_page=10" HYPERLINK "http://localhost:8082/listusers?page=1 HYPERLINK "http://localhost:8082/listusers?page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listusers?page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/listusers?page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listusers?page=1&per\_page=10" HYPERLINK "http://localhost:8082/listusers?page=1 HYPERLINK "http://localhost:8082/listusers?page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listusers?page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/listusers?page=1&per\_page=10"per\_page=10](http://localhost:8082/listusers?page=1&per_page=10) Response: list of users [JSON array]

{

     "data": [

         {

             "user\_email": "prashant1.kolage@emcure.com",

             "user\_employee\_id": **null**,

             "user\_last\_name": "Kolage1",

             "user\_username": "prashant.kolage1",

             "user\_designation\_id": 0,

             "user\_address": "Pune",

             "user\_approval\_status": "1",

             "user\_report\_to": 0,

             "user\_department\_id": 3,

             "user\_role\_name": "Executor",

             "user\_added\_by": 1,

             "user\_role\_id": 1,

             "user\_first\_name": "Prashant1",

             "user\_enable\_status": "1",

             "user\_id": 1,

             "user\_mobile": "9011997809",

             "user\_organization\_id": 1,

             "user\_default\_password\_changed": "0",

             "user\_userpassword": "$2a$10$2VQmeQGBsQIYuFFQY4v5i6Hl1KEROJ/Rp5gS",

             "user\_location\_id": 1

         }

     ],

     "meta": {

         "path": "<http://localhost:8082/listusers>",

         "per\_page": 10,

         "total": 2,

         "last\_page": 1,

         "from": 1,

         "to": 10,

         "current\_page": 1

     },

     "links": {

         "next": "<http://localhost:8082/listusers?page=1>",

         "last": "<http://localhost:8082/listusers?page=1>",

         "prev": "<http://localhost:8082/listusers?page=1>",

         "first": "<http://localhost:8082/listusers?page=1>"

     }

}

Description: get users by admin access Function : UsersController.listUsers()

* PATH: /authenticateUser Method: POST

Input: <http://localhost:9002/authenticateUser>

Response: JSON String

{

     "response": {

       "data": {

            "sess\_role\_id": 1,

             "sess\_user\_email": "prashant.kolage@emcure.com",

             "user\_full\_name": "Prashant Kolage",

             "sess\_user\_id": 1,

             "sess\_user\_mobile": "9011997809",

             "sess\_user\_default\_password": "1"

         },

         "message": "Authenticated",

         "status": "Success",

         "token": "eyJhbGciOiJIUzUxMiJ9.eyJzdWIiOiJwcmFzaGFudC5rb2xhZ2UiLCJpYXQiOjE2MTc3ODgzMjQs

mV4cCI6MTYxNzc5MTkyNH0.I2EHy6ExzQDUpHvA\_822AAON4jkoZLovP\_S

LV4jvE3XdjGQ5CfYdQsHue9ypno5toU7tbttoGF6umu3wYN0eg"

     }

}

Description: authenticate user login cred Function : UsersController.authenticateUser()

* PATH: /forgotpassword Method: POST

Input: [http://localhost:8082/forgotpassword?user\_username=prashant.kolage1 HYPERLINK "http://localhost:8082/forgotpassword?user\_username=prashant.kolage1&user\_email=prashantkolage17@gmail.com" HYPERLINK "http://localhost:8082/forgotpassword?user\_username=prashant.kolage1 HYPERLINK "http://localhost:8082/forgotpassword?user\_username=prashant.kolage1&user\_email=prashantkolage17@gmail.com"& HYPERLINK "http://localhost:8082/forgotpassword?user\_username=prashant.kolage1&user\_email=prashantkolage17@gmail.com"user\_email=prashantkolage17@gmail.com" HYPERLINK "http://localhost:8082/forgotpassword?user\_username=prashant.kolage1&user\_email=prashantkolage17@gmail.com"& HYPERLINK "http://localhost:8082/forgotpassword?user\_username=prashant.kolage1&user\_email=prashantkolage17@gmail.com" HYPERLINK "http://localhost:8082/forgotpassword?user\_username=prashant.kolage1 HYPERLINK "http://localhost:8082/forgotpassword?user\_username=prashant.kolage1&user\_email=prashantkolage17@gmail.com"& HYPERLINK "http://localhost:8082/forgotpassword?user\_username=prashant.kolage1&user\_email=prashantkolage17@gmail.com"user\_email=prashantkolage17@gmail.com" HYPERLINK "http://localhost:8082/forgotpassword?user\_username=prashant.kolage1&user\_email=prashantkolage17@gmail.com"user\_email=prashantkolage17@gmail.com](http://localhost:8082/forgotpassword?user_username=prashant.kolage1&user_email=prashantkolage17@gmail.com)

Response: JSON String

{

     "response": {

         "message": "Password reset done",

         "status": "Success"

     }

}

Description: Send Reset password to registered email Function : UsersController.forgotPassword()

* PATH: /saveUser Method: POST

Input: <http://127.0.0.1:8082/saveUser>

{

                "user\_email": "prashant1.kolage@emcure.com",

                "user\_employee\_id": **null**,

                "user\_last\_name": "Kolage1",

                "user\_username": "prashant.kolage1",

                "user\_designation\_id": 0,

                "user\_address": "Pune",

                "user\_approval\_status": "1",

                "user\_report\_to": 0,

                "user\_department\_id": 3,

                "user\_added\_by": 1,

                "user\_role\_id": 1,

                "user\_first\_name": "Prashant1",

                "user\_enable\_status": "1",

                "user\_id": 1,

                "user\_mobile": "9011997809",

                "user\_organization\_id": 1,

                "user\_default\_password\_changed": "1",

                "user\_userpassword": "Kolage",

                "user\_location\_id": 1

             }

Response: JSON String

{

     "response": {

         "data": {

             "user\_name": "prashant.kolage1"

         },

         "message": "User created successfully",

         "status": "Success"

     }

}

Description: register/save user Function : UsersController.saveUser()

* PATH: /editUser Method: POST

Input: <http://127.0.0.1:8082/editUser?user_id=1>

Response: JSON String

{

     "response": {

         "data": {

             "user\_data": {

                  "user\_email": "prashant.kolage@emcure.com",

                 "user\_employee\_id": **null**,

                 "user\_last\_name": "Kolage",

                 "user\_username": "prashant.kolage",

                  "user\_designation\_id": 0,

                 "user\_address": "Pune",

                 "user\_approval\_status": "1",

                 "user\_report\_to": 0,

                 "user\_department\_id": 3,

                 "user\_added\_by": 1,

                 "user\_role\_id": 1,

                 "user\_first\_name": "Prashant",

                 "user\_enable\_status": "1",

                 "user\_id": 1,

                 "user\_mobile": "9011997809",

                 "user\_organization\_id": 1,

                 "user\_default\_password\_changed": "1",

                "user\_userpassword": "$2a$10$uBRS0OZiMEwLGPDy8l0FgO",

                 "user\_location\_id": 1

             }

         },

         "status": "Success"

     }

Description: edit user

Function : UsersController.editUser()

* PATH: /updateUser Method: POST

Input: <http://127.0.0.1:8082/updateUser?user_id=1>

Response: JSON String

{

     "response": {

         "data": {

             "user\_name": "prashant.kolage"

         },

         "message": "User updated successfully",

         "status": "Success"

     }

}

Description: update existing user Function : UsersController.updateUser()

* PATH: /isusernameexists Method: POST

Input: [http://127.0.0.1:8082/isusernameexists?user\_id=1 HYPERLINK "http://127.0.0.1:8082/isusernameexists?user\_id=1&user\_name=ram" HYPERLINK "http://127.0.0.1:8082/isusernameexists?user\_id=1 HYPERLINK "http://127.0.0.1:8082/isusernameexists?user\_id=1&user\_name=ram"& HYPERLINK "http://127.0.0.1:8082/isusernameexists?user\_id=1&user\_name=ram"user\_name=ram" HYPERLINK "http://127.0.0.1:8082/isusernameexists?user\_id=1&user\_name=ram"& HYPERLINK "http://127.0.0.1:8082/isusernameexists?user\_id=1&user\_name=ram" HYPERLINK "http://127.0.0.1:8082/isusernameexists?user\_id=1 HYPERLINK "http://127.0.0.1:8082/isusernameexists?user\_id=1&user\_name=ram"& HYPERLINK "http://127.0.0.1:8082/isusernameexists?user\_id=1&user\_name=ram"user\_name=ram" HYPERLINK "http://127.0.0.1:8082/isusernameexists?user\_id=1&user\_name=ram"user\_name=ram](http://127.0.0.1:8082/isusernameexists?user_id=1&user_name=ram)

Response: JSON String

{

     "response": {

         "message": "Username does exist",

         "status": "Success"

     }

}

Description: checks if username exists Function : UsersController.isUserNameExists()

* PATH: / changepassword

Method: POST

Input: [http://localhost:8082/changepassword?user\_old\_password=Prashant1 HYPERLINK "http://localhost:8082/changepassword?user\_old\_password=Prashant1&user\_new\_password=ram.kolage" HYPERLINK "http://localhost:8082/changepassword?user\_old\_password=Prashant1 HYPERLINK "http://localhost:8082/changepassword?user\_old\_password=Prashant1&user\_new\_password=ram.kolage"& HYPERLINK "http://localhost:8082/changepassword?user\_old\_password=Prashant1&user\_new\_password=ram.kolage"user\_new\_password=ram.kolage" HYPERLINK "http://localhost:8082/changepassword?user\_old\_password=Prashant1&user\_new\_password=ram.kolage"& HYPERLINK "http://localhost:8082/changepassword?user\_old\_password=Prashant1&user\_new\_password=ram.kolage" HYPERLINK "http://localhost:8082/changepassword?user\_old\_password=Prashant1 HYPERLINK "http://localhost:8082/changepassword?user\_old\_password=Prashant1&user\_new\_password=ram.kolage"& HYPERLINK "http://localhost:8082/changepassword?user\_old\_password=Prashant1&user\_new\_password=ram.kolage"user\_new\_password=ram.kolage" HYPERLINK "http://localhost:8082/changepassword?user\_old\_password=Prashant1&user\_new\_password=ram.kolage"user\_new\_password=ram.kolage](http://localhost:8082/changepassword?user_old_password=Prashant1&user_new_password=ram.kolage)

Response: JSON String

{

     "response": {

         "message": "Password Changed",

         "status": "Success"

     }

Description: change user's password Function : UsersController.changepassword()

* PATH: /saveuseraccess Method: POST

Input: <http://localhost:8082/saveuseraccess>

{

     "umap\_user\_id": 1,

     "mapping\_list" :

        [

{

           "umap\_orga\_id":1,

          "umap\_loca\_id":1,

          "umap\_dept\_id":3

         }

]

}

Response: JSON String

{

     "response": {

         "message": "User access is saved",

         "status": "Success"

      }

}

Description: maps user entity mapping/save user entity

Function : UsersController.saveUserAccess()

* PATH: /getuseraccess Method: POST

Input: <http://127.0.0.1:8082/getuseraccess?user_id=1>

Response: JSON String

{

     "response": {

         "data": {

             "access\_remaining": [],

             "access\_set": [

                 {

                     "loca\_id": 1,

                     "orga\_name": "NA",

                    "user\_id": 1,

                     "loca\_name": "Hinjewadi EOU",

                     "umap\_id": 1,

                     "orga\_id": 1,

                     "dept\_name": "HR",

                     "dept\_id": 3

                 }

             ]

         },

          "status": "Success"

     }

}

Description: get list of user entity mapping access Function : UsersController.getUserAccess()

* PATH: /enabledisableuser Method: POST

Input: /enabledisableuser?user\_enable\_status=1&user\_id=1

Response: JSON String

{

     "response": {

         "message": "User enabled successfully",

         "status": "Success"

     }

}

Description: Enable Or Disable user

Function : UsersController.enableDisableUser()

* PATH: /userlogout Method: POST

Input: <http://127.0.0.1:8082/userlogout>

Response: JSON String

{

     "response": {

           "message": "Logged out",

         "status": "Success"

      }

}

Description: invalidate session and log out user Function : UsersController.userLogout()

* PATH: /myaccount Method: POST

Input : <http://localhost:8082/myaccount>

Response: JSON String

{

     "response": {

         "data": {

             "user\_details": {

                 "user\_id": 2,

                 "user\_username": "ram.kolage",

                 "user\_userpassword": "$2a$10$Ugj0npuKGr7EKJZUbX7X5O",

                 "user\_employee\_id": **null**,

                 "user\_first\_name": "Ram",

                 "user\_last\_name": "Kolage",

                 "user\_mobile": "9370471472",

                 "user\_email": "ram@emcure.com",

                 "user\_address": "Pune",

                 "user\_role\_id": 1,

                 "user\_organization\_id": 1,

                 "user\_location\_id": 1,

                 "user\_department\_id": 3,

                  "user\_designation\_id": 0,

                 "user\_report\_to": 0,

                 "user\_enable\_status": "1",

                 "user\_approval\_status": "1",

                 "user\_added\_by": 1,

                 "user\_created\_at": "17-06-2019 11:01:39",

                 "profile\_pic": **null**,

                 "user\_default\_password\_changed": "1"

             }

         },

         "status": "Success"

     }

}

Description: gets the user details Function : UsersController.myAccount()

* PATH: /downloaduserlist Method: POST

Input: [http://localhost:8082/downloaduserlist?page=1 HYPERLINK "http://localhost:8082/downloaduserlist?page=1&per\_page=10&pagination=true" HYPERLINK "http://localhost:8082/downloaduserlist?page=1 HYPERLINK "http://localhost:8082/downloaduserlist?page=1&per\_page=10&pagination=true"& HYPERLINK "http://localhost:8082/downloaduserlist?page=1&per\_page=10&pagination=true"per\_page=10 HYPERLINK "http://localhost:8082/downloaduserlist?page=1&per\_page=10&pagination=true"& HYPERLINK "http://localhost:8082/downloaduserlist?page=1&per\_page=10&pagination=true"pagination=true" HYPERLINK "http://localhost:8082/downloaduserlist?page=1&per\_page=10&pagination=true"& HYPERLINK "http://localhost:8082/downloaduserlist?page=1&per\_page=10&pagination=true" HYPERLINK "http://localhost:8082/downloaduserlist?page=1 HYPERLINK "http://localhost:8082/downloaduserlist?page=1&per\_page=10&pagination=true"& HYPERLINK "http://localhost:8082/downloaduserlist?page=1&per\_page=10&pagination=true"per\_page=10 HYPERLINK "http://localhost:8082/downloaduserlist?page=1&per\_page=10&pagination=true"& HYPERLINK "http://localhost:8082/downloaduserlist?page=1&per\_page=10&pagination=true"pagination=true" HYPERLINK "http://localhost:8082/downloaduserlist?page=1&per\_page=10&pagination=true"per\_page=10 HYPERLINK "http://localhost:8082/downloaduserlist?page=1&per\_page=10&pagination=true" HYPERLINK "http://localhost:8082/downloaduserlist?page=1 HYPERLINK "http://localhost:8082/downloaduserlist?page=1&per\_page=10&pagination=true"& HYPERLINK "http://localhost:8082/downloaduserlist?page=1&per\_page=10&pagination=true"per\_page=10 HYPERLINK "http://localhost:8082/downloaduserlist?page=1&per\_page=10&pagination=true"& HYPERLINK "http://localhost:8082/downloaduserlist?page=1&per\_page=10&pagination=true"pagination=true" HYPERLINK "http://localhost:8082/downloaduserlist?page=1&per\_page=10&pagination=true"& HYPERLINK "http://localhost:8082/downloaduserlist?page=1&per\_page=10&pagination=true" HYPERLINK "http://localhost:8082/downloaduserlist?page=1 HYPERLINK "http://localhost:8082/downloaduserlist?page=1&per\_page=10&pagination=true"& HYPERLINK "http://localhost:8082/downloaduserlist?page=1&per\_page=10&pagination=true"per\_page=10 HYPERLINK "http://localhost:8082/downloaduserlist?page=1&per\_page=10&pagination=true"& HYPERLINK "http://localhost:8082/downloaduserlist?page=1&per\_page=10&pagination=true"pagination=true" HYPERLINK "http://localhost:8082/downloaduserlist?page=1&per\_page=10&pagination=true"pagination=true](http://localhost:8082/downloaduserlist?page=1&per_page=10&pagination=true)

Response: JSON String

{

     "response": {

         "data": {

             "user\_list": [

                 {

                     "user\_email": "prashantkolage17@gmail.com",

                     "orga\_name": "NA",

                     "user\_username": "prashant.kolage1",

                     "loca\_name": "Hinjewadi EOU",

                     "dept\_name": "HR",

                     "user\_fullname": "Prashant1 Kolage1",

                     "user\_role\_name": "Executor"

                 },

                 {

                     "user\_email": "ram@emcure.com",

                     "orga\_name": "NA",

                     "user\_username": "ram.kolage",

                     "loca\_name": "Hinjewadi EOU",

                     "dept\_name": "HR",

                     "user\_fullname": "Ram Kolage",

                     "user\_role\_name": "Executor"

                 }

             ]

         },

         "meta": {

             "path": "<http://localhost:8082/downloaduserlist>",

             "per\_page": 10,

             "total": 2,

             "last\_page": 1,

             "from": 1,

             "to": 10,

             "current\_page": 1

         },

         "links": {

             "next": "<http://localhost:8082/downloaduserlist?page=1>",

             "last": "<http://localhost:8082/downloaduserlist?page=1>",

             "prev": "<http://localhost:8082/downloaduserlist?page=1>",

             "first": "<http://localhost:8082/downloaduserlist?page=1>"

         },

         "status": "Success"

     }

}

Description: gets the user list

Function : UsersController.downloadUserList()

* PATH: /sendcredentialsmail Method: POST

Input: <http://127.0.0.1:8082/sendcredentialsmail?user_id=1>

Response: JSON String

{

     "response": {

          "message": "Credentials Sent",

         "status": "Success"

     }

}

Description: sends your credentials

Function : UsersController.sendcredentialsmail()

* PATH: /removeUserAccess Method: POST

Input: <http://localhost:8082/removeUserAccess>

{

     "umap\_user\_id": 3,

     "umap\_ids" :

        [{

           "umap\_id" : 2,

          "umap\_orga\_id":1,

          "umap\_loca\_id":1,

          "umap\_dept\_id":3

         }]

}

Response: JSON String

{

     "response": {

         "message": "Removed user access",

         "status": "Success"

     }

}

Description: removes user access

Function : UsersController.removeUserAccess()

* PATH: /usermappinglist Method: POST

Input: [http://localhost:8082/usermappinglist?pagination=true HYPERLINK "http://localhost:8082/usermappinglist?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/usermappinglist?pagination=true HYPERLINK "http://localhost:8082/usermappinglist?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/usermappinglist?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/usermappinglist?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/usermappinglist?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/usermappinglist?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/usermappinglist?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/usermappinglist?pagination=true HYPERLINK "http://localhost:8082/usermappinglist?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/usermappinglist?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/usermappinglist?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/usermappinglist?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/usermappinglist?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/usermappinglist?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/usermappinglist?pagination=true HYPERLINK "http://localhost:8082/usermappinglist?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/usermappinglist?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/usermappinglist?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/usermappinglist?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/usermappinglist?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/usermappinglist?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/usermappinglist?pagination=true HYPERLINK "http://localhost:8082/usermappinglist?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/usermappinglist?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/usermappinglist?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/usermappinglist?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/usermappinglist?pagination=true&page=1&per\_page=10"per\_page=10](http://localhost:8082/usermappinglist?pagination=true&page=1&per_page=10)

Response: JSON String

{

     "response": {

         "data": {

             "orgonogram": [

                 {

                     "Unit\_Heads": [],

                     "Function\_Heads": [],

                     "orga\_name": "NA",

                     "loca\_name": "Hinjewadi EOU",

                     "dept\_name": "Stores",

                     "Evaluators": [],

                     "Executors": [],

                     "Entity\_Heads": []

                 },

                 {

                     "Unit\_Heads": [],

                     "Function\_Heads": [],

                     "orga\_name": "Emcure Pharmaceuticals Limited",

                     "loca\_name": "Hinjewadi EOU",

                     "dept\_name": "HR",

                     "Evaluators": [],

                     "Executors": [

                         {

                            "user\_id": 1,

                            "user\_full\_name": "Prashant1 Kolage1"

                         }

                     ],

                     "Entity\_Heads": []

                 }

             ]

         },

         "meta": {

             "path": "<http://localhost:8082/usermappinglist>",

             "per\_page": 10,

             "total": 2,

             "last\_page": 1,

             "from": 1,

             "to": 10,

             "current\_page": 1

         },

         "links": {

             "next": "<http://localhost:8082/usermappinglist?page=1>",

             "last": "<http://localhost:8082/usermappinglist?page=1>",

             "prev": "<http://localhost:8082/usermappinglist?page=1>",

             "first": "<http://localhost:8082/usermappinglist?page=1>"

         },

         "status": "Success"

     }

}

Description: get user mapping list

Function : UsersController.usermappinglist()

* PATH: /getuserwithaccessforcommonemail Method: POST

Input: <http://127.0.0.1:8082/getuserwithaccessforcommonemail>

Response: JSON String

{

     "response": {

         "data": {

             "user\_access\_for\_common\_email": {

                 "Entity": [

                     {

                         "orga\_name": "NA",

                         "orga\_id": 1

                     }

                 ],

                 "Function": [

                     {

                         "loca\_id": 1,

                         "orga\_id": 1,

                         "dept\_name": "HR",

                         "dept\_id": 3

                     }

                 ],

                 "Users": [

                     {

                         "loca\_id": 1,

                         "user\_email": "prashant1.kolage@emcure.com",

                         "user\_id": 1,

                         "user\_name": "Prashant1 Kolage1",

                         "orga\_id": 1,

                         "dept\_id": 3

                     }

                 ],

                 "Unit": [

                     {

                         "loca\_id": 1,

                         "loca\_name": "Hinjewadi EOU",

                         "orga\_id": 1

                     }

                 ]

             }

         },

         "status": "Success"

     }

}

Description: get user having common email

Function : UsersController.getUserWithAccessForCommonEmail()

* PATH: /importusersfromfile Method: POST

Input: <http://localhost:8082/importusersfromfile>

Response: JSON String

{

     "response": {

         "data": {

             "addedTasks": [

                 {

                     "name": "tushar.more"

                 }

             ],

             "neglectedTasks": []

         },

          "status": "Success"

     }

}

Description: import user / complaince proof against user Function : UsersController.importLegalUpdates()

* PATH: /listUnits Method: GET

Input: /listUnits?pagination=true&page=1&per\_page=10

Response: JSON Array String

{

     "data": [

         {

             "loca\_id": "1",

             "loca\_name": "Hinjewadi EOU"

         }

     ],

     "meta": {

         "path": "<http://localhost:8082/units>",

         "per\_page": 10,

         "total": 1,

         "last\_page": 1,

         "from": 1,

         "to": 10,

         "current\_page": 1

     },

     "links": {

         "next": "<http://localhost:8082/units?page=1>",

         "last": "<http://localhost:8082/units?page=1>",

         "prev": "<http://localhost:8082/units?page=1>",

         "first": "<http://localhost:8082/units?page=1>"

     }

}

Description: get list of units against the user id Function : UnitController.listUnits()

* PATH: /listallunits Method: GET

Input: [http://localhost:8082/listallunits?pagination=true HYPERLINK "http://localhost:8082/listallunits?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/listallunits?pagination=true HYPERLINK "http://localhost:8082/listallunits?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listallunits?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/listallunits?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listallunits?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/listallunits?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listallunits?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/listallunits?pagination=true HYPERLINK "http://localhost:8082/listallunits?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listallunits?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/listallunits?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listallunits?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/listallunits?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/listallunits?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/listallunits?pagination=true HYPERLINK "http://localhost:8082/listallunits?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listallunits?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/listallunits?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listallunits?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/listallunits?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listallunits?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/listallunits?pagination=true HYPERLINK "http://localhost:8082/listallunits?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listallunits?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/listallunits?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listallunits?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/listallunits?pagination=true&page=1&per\_page=10"per\_page=10](http://localhost:8082/listallunits?pagination=true&page=1&per_page=10)

Response: JSON String

{

     "data": [

          {

             "loca\_id": "1",

              "loca\_name": "Hinjewadi EOU"

          },

         {

             "loca\_id": "2",

              "loca\_name": "Hinjewadi ARCH"

          }

     ],

     "meta": {

         "path": "<http://127.0.0.1:8082/listallunits>",

         "per\_page": 5,

         "total": 2,

          "last\_page": 1,

          "from": 1,

         "to": 5,

         "current\_page": 1

      },

     "links": {

         "next": "<http://127.0.0.1:8082/listallunits?page=1>",

         "last": "<http://127.0.0.1:8082/listallunits?page=1>",

         "prev": "<http://127.0.0.1:8082/listallunits?page=1>",

         "first": "<http://127.0.0.1:8082/listallunits?page=1>"

     }

}

Description: gets the list of all units

Function : UnitController.listAllUnits()

* PATH: /saveunit Method: **POST**

Input : <http://127.0.0.1:8082/saveunit?loca_name=pune>

Response: JSON String

{

     "response": {

         "data": {

             "loca\_name": "pune"

         },

         "message": "Unit Saved",

         "status": "Success"

     }

}

Description: saves the unit Function : UnitController.saveUnit()

* PATH: /updateunit Method: POST

Input: <http://127.0.0.1:8082/updateunit/2?loca_name=Pune>

Response: JSON String

{

     "response": {

         "data": {

             "unit\_data": {

                 "loca\_id": 2,

                 "loca\_name": "Pune"

             }

         },

         "message": "Unit Updated",

         "status": "Success"

     }

}

Description: updates the unit Function : UnitController.updateUnit()

* PATH: /isLocaNameExist Method: GET

Input : [http://127.0.0.1:8082/isLocaNameExist?loca\_name=pune HYPERLINK "http://127.0.0.1:8082/isLocaNameExist?loca\_name=pune&loca\_id=1" HYPERLINK "http://127.0.0.1:8082/isLocaNameExist?loca\_name=pune HYPERLINK "http://127.0.0.1:8082/isLocaNameExist?loca\_name=pune&loca\_id=1"& HYPERLINK "http://127.0.0.1:8082/isLocaNameExist?loca\_name=pune&loca\_id=1"loca\_id=1" HYPERLINK "http://127.0.0.1:8082/isLocaNameExist?loca\_name=pune&loca\_id=1"& HYPERLINK "http://127.0.0.1:8082/isLocaNameExist?loca\_name=pune&loca\_id=1" HYPERLINK "http://127.0.0.1:8082/isLocaNameExist?loca\_name=pune HYPERLINK "http://127.0.0.1:8082/isLocaNameExist?loca\_name=pune&loca\_id=1"& HYPERLINK "http://127.0.0.1:8082/isLocaNameExist?loca\_name=pune&loca\_id=1"loca\_id=1" HYPERLINK "http://127.0.0.1:8082/isLocaNameExist?loca\_name=pune&loca\_id=1"loca\_id=1](http://127.0.0.1:8082/isLocaNameExist?loca_name=pune&loca_id=1)

Response: JSON String

{

     "response": {

         "data": {

             "is\_exists": **false**

         },

         "status": "Success"

     }

}

Description: checks if the unit is exists by id Function : UnitController.isLocaNameExist()

* PATH: /importunit Method: POST Response: JSON String

{

     "response": {

         "status": "Success"

     }

}

Description: import unit

Function : UnitController.importLegalUpdates()

---------------------------------------------------------------------

--------------------------------------------------------

* PATH: /getSupportQuery Method: POST

Input: <http://localhost:8082/getSupportQuery>

{

"Cname":"Prashant Kolage",

"Cemail":"prashantkolage17@gmail.com",

"Cmobile":"9011997809",

"Cmessage":"getSupportQuery email Testing"

}

Response: JSON String

{

     "response": {

         "message": "Email Sent",

         "status": "Success"

     }

}



Description: sends the support email

Function : TechSupportController.getSupportQuery()

* PATH: /sendcommonemail Method: POST

Input: <http://localhost:8082/sendcommonemail>

{

"email\_ids":["prashantkolage17@gmail.com"],

"subject":"Commmom email",

"body":"Hii

Please find below attachment"

}

Response: JSON String

{

     "response": {

         "message": "Email Sent",

         "status": "Success"

     }

}



Description: sends email for all emails

Function : TechSupportController.sendCommonEmail()

---------------------------------------------------------------------

---------------------------------------------------------------------

* PATH: /savetasksusermapping

Method: POST

Input: <http://localhost:8082/savetasksusermapping>

JsonString:

{

     "dept\_id": 3,

     "orga\_id": 2,

     "pr\_user\_id": 1,

     "rw\_user\_id": 1,

     "fh\_user\_id": 1,

     "tasks\_list":[{

         "tmap\_task\_id": 2,

         "tmap\_lexcare\_task\_id":"L0005675"

     }],

     "loca\_list":[{

         "tmap\_loca\_id":2

     }]

}

Response: JSON String

{

     "response": {

         "data": {

             "assigned\_task\_mapping\_data": [

                 {

                     "client\_tasks\_id": "D02030000002"

                  }

             ]

          },

         "status": "Success"

      }

}

Description: saves the task user mapping entity

Function : TasksUserMappingController.savetasksusermapping()

* PATH: /getdistinctcountries Method: GET

Input: <http://127.0.0.1:8082/getdistinctcountries>

Response: JSON String array

{

     "response": {

         "data": {

             "countries": [

                 {

                     "country\_name": "India",

                     "country\_id": 1

                 }

              ]

          },

         "status": "Success"

     }

}

Description: get list of uniq countries

Function : TasksUserMappingController.getDistinctCountries()

* PATH: /getallstateforcountry Method: POST

Input: <http://127.0.0.1:8082/getallstateforcountry/1>

Response: JSON String array

{

     "response": {

         "data": {

             "state\_list": [

                 {

                     "state\_name": "MH",

                     "state\_id": "1"

                 }

             ]

         },

         "status": "Success"

     }

}

Description: get list of states of country

Function : TasksUserMappingController.getStatesForCountry()

* PATH: /getallcatlaw Method: POST

Input : /getallcatlaw?country\_id=1&state\_id=2&searching\_for=tasksmapping

Response: JSON String array

{

     "response": {

         "data": {

             "cat\_law\_list": [

                 {

                     "cat\_law\_name": "Regulatory",

                     "cat\_law\_id": "7"

                 },

{

"cat\_law\_name": "Labour Laws",

"cat\_law\_id": "5"

}

             ]

         },

         "status": "Success"

     }

}

Description: get list of category law list

Function: TasksUserMappingController.getAllCatLawFromMst\_task()

* PATH: /getalllegi Method: POST

Input: /getalllegi/7?country\_id=1&state\_id=1&searching\_for=tasksmapping

Response: JSON String array

{

     "response": {

         "data": {

             "legi\_list": [

                 {

                     "legi\_id": "58",

                     "legi\_name": "The Electricity Act, 2003"

                 }

             ]

         },

         "status": "Success"

     }

}

Description: get list of all legislations

Function : TasksUserMappingController.getAllLegiFromMst\_task()

* PATH: /getalllegirule Method: POST

Input : compliance/getalllegirule/58?cat\_id=7&country\_id=1&state\_id=2 Response: JSON String array

{

"response": {

"data": {

"legi\_rule\_list": [

{

"rule\_id": "821",

"rule\_name": "The Intimation of Accidents (Form and Time of service of notice) Rules, 2005"

},

{

"rule\_id": "819",

"rule\_name": " The Electricity Rules, 2005"

}

]

},

"status": "Success"

}

}

Description: get list of all legislations rule Function :

TasksUserMappingController.getAllLegiruleFromMst\_task()

* PATH: /searchtasksforusermapping Method: POST

Input : [/compliance/searchtasksforusermapping](http://localhost:8082/compliance/searchtasksforusermapping)

{

     "dataRequiredFor": "taskMapping",

      "country\_id":1,

     "state\_id": 1,

     "cat\_id": 7,

     "legi\_id": 58,

     "rule\_id": 1059,

     "orga\_id":2,

     "loca\_id": 2

     }

Response: JSON String array

{

     "response": {

         "data": {

             "allTasks": [

                 {

                     "task\_impact": "Major",

                     "task\_rule\_name": "The Central Electricity Regulation 2011",

                     "task\_reference": "Regulation 5",

                     "task\_who": "Owner",

                     "task\_activity": "Provide a detailed Safety Manual",

                     "task\_legi\_name": "The Electricity Act, 2003",

                     "task\_id": "1",

                     "task\_procedure": "NA",

                     "lexcare\_task\_id": "C0011121",

                     "task\_when": "On applicability of the act",

                     "task\_frequency": "One\_Time"

                 }

             ],

             "totalNoOfTasks": 1,

             "assignDropDowns": [

                 {

                     "Entity": [

                         {

                            "orga\_name": "NA",

                            "orga\_id": 1

                         },

                         {

                            "orga\_name": "Emcure Pharmaceuticals Limited",

                            "orga\_id": 2

                         }

                     ],

                     "Function": [

                         {

                             "loca\_id": 1,

                             "orga\_id": 1,

                             "dept\_name": "HR",

                             "dept\_id": 3

                         },

                         {

                             "loca\_id": 1,

                             "orga\_id": 1,

                             "dept\_name": "HR",

                             "dept\_id": 3

                         },

                         {

                             "loca\_id": 1,

                             "orga\_id": 2,

                             "dept\_name": "HR",

                             "dept\_id": 3

                         }

                     ],

                     "Executor": [

                         {

                              "loca\_id": 1,

                              "orga\_id": 1,

                              "dept\_name": "HR",

                             "dept\_id": 3

                         },

                         {

                             "loca\_id": 1,

                             "orga\_id": 1,

                             "dept\_name": "HR",

                             "dept\_id": 3

                         },

                         {

                             "loca\_id": 1,

                             "orga\_id": 2,

                             "dept\_name": "HR",

                             "dept\_id": 3

                         }

                     ],

                    "Function\_Head": [],

                     "Unit": [

                         {

                             "loca\_id": 1,

                             "loca\_name": "Hinjewadi EOU",

                             "orga\_id": 1

                         },

                         {

                             "loca\_id": 1,

                             "loca\_name": "Hinjewadi EOU",

                             "orga\_id": 2

                         }

                     ],

                     "Evaluator": []

                 }

             ]

         },

         "status": "Success"

     }

}

Description: returns list of taks searched for

Function :

TasksUserMappingController.searchTasksForTaskUserMapping()

35] PATH: /getallmappedtasksforenablingpage

Method: POST

Input : [http://localhost:8082/getallmappedtasksforenablingpage?pagination=true HYPERLINK "http://localhost:8082/getallmappedtasksforenablingpage?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/getallmappedtasksforenablingpage?pagination=true HYPERLINK "http://localhost:8082/getallmappedtasksforenablingpage?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getallmappedtasksforenablingpage?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/getallmappedtasksforenablingpage?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getallmappedtasksforenablingpage?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/getallmappedtasksforenablingpage?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getallmappedtasksforenablingpage?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/getallmappedtasksforenablingpage?pagination=true HYPERLINK "http://localhost:8082/getallmappedtasksforenablingpage?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getallmappedtasksforenablingpage?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/getallmappedtasksforenablingpage?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getallmappedtasksforenablingpage?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/getallmappedtasksforenablingpage?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/getallmappedtasksforenablingpage?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/getallmappedtasksforenablingpage?pagination=true HYPERLINK "http://localhost:8082/getallmappedtasksforenablingpage?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getallmappedtasksforenablingpage?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/getallmappedtasksforenablingpage?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getallmappedtasksforenablingpage?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/getallmappedtasksforenablingpage?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getallmappedtasksforenablingpage?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/getallmappedtasksforenablingpage?pagination=true HYPERLINK "http://localhost:8082/getallmappedtasksforenablingpage?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getallmappedtasksforenablingpage?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/getallmappedtasksforenablingpage?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getallmappedtasksforenablingpage?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/getallmappedtasksforenablingpage?pagination=true&page=1&per\_page=10"per\_page=10](http://localhost:8082/getallmappedtasksforenablingpage?pagination=true&page=1&per_page=10)

Response: JSON String array

{

    "response": {

        "data": {

            "Filters": [

                {

                    "Entity": [

                        {

                            "orga\_name": "Emcure Pharmaceuticals Limited",

                            "orga\_id": 2

                        }

                    ],

                    "Function": [

                        {

                            "loca\_id": 2,

                            "orga\_id": 2,

                            "dept\_name": "HR",

                            "dept\_id": 3

                        }

                    ],

                    "FHUsers": [

                        {

                            "loca\_id": 2,

                            "fh\_id": 1,

                            "orga\_id": 2,

                            "fh\_name": "Prashant1 Kolage1",

                            "dept\_id": 3

                        }

                    ],

                    "Legislations": [

                        {

                            "task\_legi\_id": 58,

                            "loca\_id": 2,

                            "orga\_id": 2,

                            "task\_legi\_name": "The Electricity Act, 2003",

                            "dept\_id": 3

                        }

                    ],

"Users": [

                        {

                            "exec\_id": 1,

                            "loca\_id": 2,

                            "eval\_id": 1,

                            "orga\_id": 2,

                            "exec\_name": "Prashant1 Kolage1",

                            "dept\_id": 3,

                            "eval\_name": "Prashant1 Kolage1"

                        }

                    ],

                    "Unit": [

                        {

                            "loca\_id": 2,

                            "loca\_name": "Pune",

                            "orga\_id": 2

                        }

                    ],

                    "Rules": [

                        {

                            "task\_legi\_id": 58,

                            "task\_rule\_name": "The Central Electricity Authority ",

                            "task\_rule\_id": 1059

                        }

                    ]

                }

            ],

            "All\_Tasks": [

                {

                    "task\_legi\_id": 58,

                    "task\_rule\_name": "The Central Electricity Authority",

                    "executor\_name": "Prashant1 Kolage1",

                    "task\_activity": "Provide a detailed Safety Manual   ",

                    "orga\_id": 2,

                    "task\_legi\_name": "The Electricity Act, 2003",

                    "tmap\_enable\_status": "1",

                    "tmap\_id": 1,

                    "tmap\_client\_tasks\_id": "D02030000001",

                    "task\_activity\_when": "On applicability of the act",

                    "loca\_id": 2,

                    "function\_head\_id": 1,

                    "task\_activity\_who": "Owner",

                    "task\_reference": "Regulation 5",

                    "evaluator\_id": 1,

                    "task\_lexcare\_id": "C0011121",

                    "task\_rule\_id": 1059,

                    "task\_procedure": "NA",

                    "dept\_id": 3,

                    "executor\_id": 1

                 }

             ]

         },

         "meta": {

             "path": "<http://localhost:8082/getallmappedtasksforenablingpage>",

             "per\_page": 10,

             "total": 1,

             "last\_page": 1,

             "from": 1,

             "to": 10,

             "current\_page": 1

         },

        "links": {

             "next": "<http://localhost:8082/getallmappedtasksforenablingpage?page=1>",

             "last": "<http://localhost:8082/getallmappedtasksforenablingpage?page=1>",

             "prev": "<http://localhost:8082/getallmappedtasksforenablingpage?page=1>",

             "first": "<http://localhost:8082/getallmappedtasksforenablingpage?page=1>"

         },

         "status": "Success"

     }

}

* PATH: /enablingoftasks Method: POST

Input: <http://localhost:8082/enablingoftasks>

{

     "operation\_to\_perform": "enable",

     "tasks\_list":[

{ "tmap\_id": 1 },

{"tmap\_id": 2}

]

}

Response: JSON String

{

     "response": {

         "status": "Success"

     }

}

Description: enables or disables the user mapped task Function : TasksUserMappingController.enablingOfTasks()

* PATH: /changecomplianceowner Method: POST

Input: [http://localhost:9002/compliance/changecomplianceowne](http://localhost:9002/compliance/changecomplianceowner?userID=1&userName=Ram)r

Input Json String :

{

     "orga\_id": 2,

     "loca\_id": 2,

     "dept\_id": 3,

     "pr\_user\_id": 1,

     "rw\_user\_id":1,

     "fh\_user\_id":1,

     "tasks\_list": [

{

"tmap\_id":1

}

]

}

Response: JSON String

{

     "response": {

         "status": "Success"

     }

}

Description: changes the owner of complaince

Function : TasksUserMappingController.changeComplianceOwner()

* PATH: /getTaskForChangeComplianceOwnerPage Method: POST

Input : [http://localhost:8082/getTaskForChangeComplianceOwnerPage?pagination=true HYPERLINK "http://localhost:8082/getTaskForChangeComplianceOwnerPage?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/getTaskForChangeComplianceOwnerPage?pagination=true HYPERLINK "http://localhost:8082/getTaskForChangeComplianceOwnerPage?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getTaskForChangeComplianceOwnerPage?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/getTaskForChangeComplianceOwnerPage?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getTaskForChangeComplianceOwnerPage?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/getTaskForChangeComplianceOwnerPage?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getTaskForChangeComplianceOwnerPage?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/getTaskForChangeComplianceOwnerPage?pagination=true HYPERLINK "http://localhost:8082/getTaskForChangeComplianceOwnerPage?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getTaskForChangeComplianceOwnerPage?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/getTaskForChangeComplianceOwnerPage?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getTaskForChangeComplianceOwnerPage?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/getTaskForChangeComplianceOwnerPage?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/getTaskForChangeComplianceOwnerPage?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/getTaskForChangeComplianceOwnerPage?pagination=true HYPERLINK "http://localhost:8082/getTaskForChangeComplianceOwnerPage?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getTaskForChangeComplianceOwnerPage?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/getTaskForChangeComplianceOwnerPage?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getTaskForChangeComplianceOwnerPage?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/getTaskForChangeComplianceOwnerPage?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getTaskForChangeComplianceOwnerPage?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/getTaskForChangeComplianceOwnerPage?pagination=true HYPERLINK "http://localhost:8082/getTaskForChangeComplianceOwnerPage?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getTaskForChangeComplianceOwnerPage?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/getTaskForChangeComplianceOwnerPage?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getTaskForChangeComplianceOwnerPage?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/getTaskForChangeComplianceOwnerPage?pagination=true&page=1&per\_page=10"per\_page=10](http://localhost:8082/getTaskForChangeComplianceOwnerPage?pagination=true&page=1&per_page=10)

Response: JSON String

{

     "response": {

        "data": {

             "mapped\_data\_list": {

                "Filters": [

                     {

                        "Legislations": [

                            {

                                "task\_legi\_id": 58,

                                "loca\_id": 2,

                                "orga\_id": 2,

                                "task\_legi\_name": "The Electricity Act, 2003",

                                "dept\_id": 3

                            }

                        ],

                        "Rules": [

                            {

                                "task\_legi\_id": 58,

                                "task\_rule\_name": "The Central Electricity Authority",

                                "task\_rule\_id": 1059

                            }

                        ]

                    }

                 ]

             }

         },

         "meta": {

             "path": "<http://localhost:8082/getTaskForChangeComplianceOwnerPage>",

             "per\_page": 10,

             "total": 1,

             "last\_page": 1,

             "from": 1,

             "to": 10,

             "current\_page": 1

         },

         "links": {

             "next": "<http://localhost:8082/getTaskForChangeComplianceOwnerPage?page=1>",

             "last": "<http://localhost:8082/getTaskForChangeComplianceOwnerPage?page=1>",

             "prev": "<http://localhost:8082/getTaskForChangeComplianceOwnerPage?page=1>",

             "first": "<http://localhost:8082/getTaskForChangeComplianceOwnerPage?page=1>"

         },

         "status": "Success"

     }

}

Description: gets the task for changed complaince owner page Function :

TasksUserMappingController.getTaskForChangeComplianceOwnerPage()

---------------------------------------------------------------------

--------------------------------------------------------

* PATH: /getimportedtasks Method: GET

Input : [http://localhost:8082/getimportedtasks?pagination=true HYPERLINK "http://localhost:8082/getimportedtasks?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/getimportedtasks?pagination=true HYPERLINK "http://localhost:8082/getimportedtasks?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getimportedtasks?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/getimportedtasks?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getimportedtasks?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/getimportedtasks?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getimportedtasks?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/getimportedtasks?pagination=true HYPERLINK "http://localhost:8082/getimportedtasks?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getimportedtasks?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/getimportedtasks?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getimportedtasks?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/getimportedtasks?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/getimportedtasks?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/getimportedtasks?pagination=true HYPERLINK "http://localhost:8082/getimportedtasks?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getimportedtasks?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/getimportedtasks?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getimportedtasks?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/getimportedtasks?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getimportedtasks?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/getimportedtasks?pagination=true HYPERLINK "http://localhost:8082/getimportedtasks?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getimportedtasks?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/getimportedtasks?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getimportedtasks?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/getimportedtasks?pagination=true&page=1&per\_page=10"per\_page=10](http://localhost:8082/getimportedtasks?pagination=true&page=1&per_page=10)

Response: JSON String array

{

     "response": {

         "data": {

             "imported\_tasks": [

                 {

                    "task\_id": 1,

                    "task\_activity": "Provide a detailed Safety Manual",

                    "task\_activity\_who": "Owner",

                    "task\_activity\_when": "On applicability of the act",

                    "task\_cat\_law\_id": 7,

                    "task\_cat\_law\_name": "Regulatory",

                    "task\_country\_id": 1,

                    "task\_country\_name": "India",

                    "task\_legal\_due\_date": **null**,

                    "task\_event": "NA",

                    "task\_excemption\_criteria": "NA",

                    "task\_fine\_amount": 1999.00,

                    "task\_form\_no": "NA",

                    "task\_frequency": "One\_Time",

                    "task\_impact": "Major",

                    "task\_impact\_on\_organization": "Moderate",

                    "task\_impact\_on\_unit": "Major",

                    "task\_implication": "Penalty upto one lakh rupees for each contravention",

                    "task\_imprisonment\_duration": "NA",

                    "task\_imprisonment\_implies\_to": "NA",

                    "task\_interlinkage": "NA",

                    "task\_legi\_id": 58,

                    "task\_legi\_name": "The Electricity Act, 2003",

                    "task\_level": "Unit level",

                    "task\_lexcare\_task\_id": "C0011121",

                    "task\_linked\_task\_id": "NA",

                    "task\_more\_info": ". For similar installations common safety manual can ",

                    "task\_procedure": "NA",

                    "task\_prohibitive": "Prescriptive",

                    "task\_document\_reference": **null**,

                    "task\_form\_format": **null**,

                    "task\_reference": "Regulation 5",

                    "task\_rule\_id": 1059,

                    "task\_rule\_name": "The Central Electricity Authority",

                    "task\_sequence": 0,

                    "due\_date": "NA",

                    "task\_specific\_due\_date": "NA",

                    "task\_state\_id": 1,

                    "task\_state\_name": "MH",

                    "task\_sub\_event": "NA",

                    "task\_subsequent\_amount\_per\_day": 6000.00,

                    "task\_task\_type\_of\_task": "Verification",

                    "task\_subtask": **null**,

                    "task\_effective\_date": **null**,

                    "task\_weblinks": "NA",

                    "task\_approval\_status": "1",

                    "task\_added\_by": 1,

                    "task\_created\_at": "25-07-2019 00:00:00",

                    "task\_enable\_status": "1",

                    "task\_statutory\_authority": **null**

                }

            ]

         },

         "meta": {

             "path": "<http://localhost:8082/getimportedtasks>",

             "per\_page": 10,

             "total": 1,

             "last\_page": 1,

             "from": 1,

             "to": 10,

             "current\_page": 1

         },

         "links": {

             "next": "<http://localhost:8082/getimportedtasks?page=1>",

             "last": "<http://localhost:8082/getimportedtasks?page=1>",

             "prev": "<http://localhost:8082/getimportedtasks?page=1>",

             "first": "<http://localhost:8082/getimportedtasks?page=1>"

         },

         "status": "Success"

     }

}

Description: get all imported tasks

Function : TasksController.getAllImportedTasks()

* PATH: /importlegalupdatesfromfile Method: POST

Response: JSON String

Description: import legal upadate activity Function : TasksController.importLegalUpdates()

* PATH: /gethistoryfortask Method: POST

Input : <http://localhost:9002/compliance/gethistoryfortask>

{

     "tmap\_client\_task\_id": "D27050000540"

}

Response: JSON String

{

     "response": {

         "task\_history": [

             {

                "ttrn\_id": 1,

                "ttrn\_performer\_name": "Prashant1 Kolage1",

                "ttrn\_legal\_due\_date": "09-10-2019",

                "client\_task\_id": "D27050000540",

                "ttrn\_rw\_due\_date": "03-10-2019",

                "ttrn\_task\_status": "Inactive",

                "ttrn\_legal\_task\_style": "",

                "ttrn\_reason\_for\_non\_compliance": "QPR submitted in SEEPZ on 25.10.2019.",

                "ttrn\_pr\_due\_date": "01-10-2019",

                "ttrn\_task\_completed\_by": "Prashant1 Kolage1",

                "ttrn\_uh\_due\_date": "07-10-2019",

                "ttrn\_performer\_comments": "QPR for Emcure EOU / SPD / Kurkumbh ",

                "ttrn\_document": "1",

                "ttrn\_legal\_task\_status": "Inactive",

                "document\_list": [],

                "ttrn\_fh\_due\_date": "24-10-2019",

                "ttrn\_completed\_date": "24-10-2019",

                "task\_frequency": **null**

             }

         ],

         "subTaskHistory": [],

         "status": "Success"

     }

}

Description: get the history of task

Function : TasksController.getHistoryForTask()

* PATH: /getdetailsfortask Method: POST

Input: <http://localhost:8082/getdetailsfortask>

{

     "tmap\_client\_task\_id": "D27050000541"

}

Response: JSON String

{

     "response": {

         "data": {

             "task\_details": {

                "specific\_due\_date": "NA",

                "form\_no": "NA",

                "activity": "Provide a detailed Safety Manual  and it should be site specific",

                "function\_head": "Tushar More",

                "prescriptive": "Prescriptive",

                "imprisonment": "NA",

                "rule": "The Central Electricity Authority",

                "when": "On applicability of the act",

                "excemption\_criteria": "NA",

                "fine\_amount\_per\_day": 6000.00,

                "frequency": "One\_Time",

                "reference": "Regulation 5",

                "interlinkage": "NA",

                "executor": "Tushar More",

                "function": "HR",

                "fine\_amount": 1999.00,

                "unit\_task\_id": "D27050000541",

                "event": "NA",

                "evaluator": "Tushar More",

                "legislation": "The Electricity Act, 2003",

                "implications": "Penalty upto one lakh rupees for each contravention and in c ",

                "level": "Unit level",

                "impact": "Major",

                "due\_date": "NA",

                "procedure": "NA",

                "linked\_task\_id": "NA",

                "unit": "Balewadi",

                "type\_of\_task": "Verification",

                "more\_information": ". For similar installations common safety manual can ",

                "impact\_on\_entity": "Moderate",

                "imprisonment\_applies\_to": "NA",

                "sub\_event": "NA",

                "weblink": "NA",

                "lexcare\_task\_id": "L0005671",

                "entity": "Emcure Pharmaceuticals Limited",

                "who": "Owner",

                "impact\_on\_unit": "Major"

             }

         },

         "status": "Success"

     }

}

Description: get task details against client task id Function : TasksController.getDetailsForTask()

* PATH: /downloadProofOfCompliance Method: GET

Input: <http://localhost:9002/compliance/downloadProofOfCompliance>

Response: attachment download

Description: download proof of complaince

Function : TasksController.downloadProofOfCompliance()

* PATH: /getdefaulttaskforinitiation Method: POST

Input: /compliance/getdefaulttaskforinitiation

Response: JSON String

{

      "response": {

         "data": {},

         "status": "Success"

     }

}

Description: get the client task id from default configuration Function : TasksController.getdefaulttaskforinitiation()

* PATH: /getalltasksforexport Method: POST

Input: <http://localhost:9002/compliance/getalltasksforexport>

Response: JSON String array

{

"response": {

"data": {

"TasksFilter": [

{

"Legislation": [],

"Sub\_Event": [],

"Type\_Of\_Task": [],

"Event": [],

"Frequency": [],

"Rule": [],

"Cat\_law": []

}

],

"OrganogramFilter": [

{

"Entity": [],

"Function": [],

"Executor": [],

"Unit": [],

"Evaluator": []

}

],

"AllTasks": [],

"TotalNoOfTasks": 0

},

"status": "Success"

}

}

Description: get the list of all tasks for export operation Function : TasksController.getAllTasksForExport()

* PATH: /gettaskformultiplecompletion Method: POST

Input:

<http://localhost:9002/compliance/gettaskformultiplecompletion>

{

"tmap\_client\_task\_id": "D33030015238",

"ttrn\_id": 1

}

Response: JSON String

{

"response": {

"data": {

"task\_list": [

{

"ttrn\_id": 50734,

"rw\_name": "sagar krios",

"client\_task\_id": "D33030015238",

"activity": "Upload electronically a consolidated return in Form 9 regarding the employees entitled to become members of the Fund showing the basic wage, retaining allowance, if any, and dearness allowance including the cash value of any food concession paid to each of such employees",

"orga\_name": "demo entity",

"rule\_name": "(i) The Employees Deposit-Linked Insurance Scheme, 1976;\n(ii)The Employees' Pension Scheme, 1995 and\n(iii) The Employees' Provident Fund Scheme, 1952",

"lexcare\_id": "L0004219",

"pr\_date": "26-12-2020",

"dept\_name": "HR",

"fh\_name": "sagar krios",

"more\_info": "NA",

"procedure": "NA",

"when": "In the case of new employees required or entitled to become members of the EPFS Fund and EPS Fund",

"ref": "Paragraph 36 (1)of EPFS and Paragraph 20(1) of EPS\n",

"legal\_date": "29-12-2021",

"loca\_name": "Emcure House",

"pr\_name": "sagar krios",

"interlink\_id": "NA",

"legi\_name": "The Employees Provident Funds and Miscellaneous Provisions Act,1952",

"who": "Employer",

"status": "Active"

}

]

},

"status": "Success"

}

}

Description: get task for performing completion operation Function : TasksController.getTaskForMultipleCompletion()

* PATH: /getalldocuments Method: POST

Input: <http://localhost:9002/compliance/getalldocuments>

Response: JSON String

{

    "response": {

        "data": {

            "TasksFilter": [

                {

                    "Legislation": [],

                    "Sub\_Event": [],

                    "Type\_Of\_Task": [],

                    "Event": [],

                    "Frequency": [],

                    "Rule": [],

                    "Cat\_law": []

                }

            ],

            "OrganogramFilter": [

                {

                    "Entity": [],

                    "Function": [],

                    "Executor": [],

                    "Unit": [],

                    "Evaluator": []

                }

            ],

            "AllTasks": [],

            "TotalNoOfTasks": 0

        },

        "status": "Success"

    }

}

Description: get all the documents for repository Function : TasksController.getAllDocumentsForRepository()

* PATH: /listofupcommingtask Method: POST

Input: <http://localhost:9002/compliance/listofupcommingtask>

Response: JSON String

{

"response": {

"data": {

"TasksFilter": [

{

"Legislation": [

{

"task\_legi\_id": 135,

"loca\_id": 34,

"orga\_id": 2,

"task\_legi\_name": "The Companies Act, 2013",

"dept\_id": 24,

"task\_cat\_law\_id": 2

},

{

"task\_legi\_id": 260,

"loca\_id": 7,

"orga\_id": 4,

"task\_legi\_name": "The Maharashtra Labour Welfare Fund Act, 1953 ",

"dept\_id": 3,

"task\_cat\_law\_id": 5

}

],

"Sub\_Event": [

{

"task\_legi\_id": 135,

"loca\_id": 34,

"task\_Sub\_Event": "Preparation of Financial Statements",

"task\_Event": "Accounts of Companies",

"orga\_id": 2,

"task\_rule\_id": 829,

"dept\_id": 24,

"task\_cat\_law\_id": 2

},{

"task\_legi\_id": 135,

"loca\_id": 37,

"task\_Sub\_Event": "Audit conducted by Auditor",

"task\_Event": "Audit and Auditors",

"orga\_id": 3,

"task\_rule\_id": 830,

"dept\_id": 24,

"task\_cat\_law\_id": 2

}

],

"Type\_Of\_Task": [

{

"task\_type\_of\_task": "Verification"

},

{

"task\_type\_of\_task": "Returns"

},

{

"task\_type\_of\_task": "Submission"

},

{

"task\_type\_of\_task": "Meetings"

},

{

"task\_type\_of\_task": "Disclosure"

},

{

"task\_type\_of\_task": "Reports"

},

{

"task\_type\_of\_task": "Payment"

},

{

"task\_type\_of\_task": "Intimation"

},

{

"task\_type\_of\_task": "Register and Records"

},

{

"task\_type\_of\_task": "Appointment"

}

],

"Event": [

{

"task\_legi\_id": 135,

"loca\_id": 34,

"task\_Event": "Accounts of Companies",

"orga\_id": 2,

"task\_rule\_id": 829,

"dept\_id": 24,

"task\_cat\_law\_id": 2

},

{

"task\_legi\_id": 135,

"loca\_id": 37,

"task\_Event": "Audit and Auditors",

"orga\_id": 3,

"task\_rule\_id": 830,

"dept\_id": 24,

"task\_cat\_law\_id": 2

}

],

"Frequency": [

{

"task\_frequency": "Yearly"

}

],

"Rule": [

{

"task\_legi\_id": 135,

"task\_rule\_name": "The Companies (Accounts) Rules, 2014",

"task\_rule\_id": 829,

"task\_cat\_law\_id": 2

},

{

"task\_legi\_id": 135,

"task\_rule\_name": "The Companies (Audit and Auditors) Rules, 2014",

"task\_rule\_id": 830,

"task\_cat\_law\_id": 2

}

],

"Cat\_law": [

{

"task\_cat\_law": "Corporate",

"task\_cat\_law\_id": 2

},

{

"task\_cat\_law": "Environment Laws",

"task\_cat\_law\_id": 3

}

]

}

],

"OrganogramFilter": [

{

"Entity": [

{

"orga\_name": "Entity Pharmaceuticals Ltd",

"orga\_id": 2

}

],

"Function": [

{

"loca\_id": 34,

"orga\_id": 2,

"dept\_name": "Secretarial",

"dept\_id": 24

},

{

"loca\_id": 7,

"orga\_id": 4,

"dept\_name": "HR",

"dept\_id": 3

}

],

"Executor": [

{

"loca\_id": 34,

"user\_id": 169,

"user\_name": "Sheetal Kulkarni",

"orga\_id": 2,

"dept\_id": 24

},{

"loca\_id": 7,

"user\_id": 96,

"user\_name": "Sanjay Salaskar",

"orga\_id": 4,

"dept\_id": 3

}

],

"Unit": [

{

"loca\_id": 34,

"loca\_name": "Emcure Pharma- Secreterial",

"orga\_id": 2

},{

"loca\_id": 7,

"loca\_name": "Zuventus MO Mumbai",

"orga\_id": 4

}

],

"Evaluator": [

{

"loca\_id": 34,

"user\_id": 316,

"user\_name": "Jayant Prakash",

"orga\_id": 2,

"dept\_id": 24

}, {

"loca\_id": 7,

"user\_id": 120,

"user\_name": "Sudeep Raj",

"orga\_id": 4,

"dept\_id": 3

}

]

}

],

"AllTasks": [

{

"task\_prohibitive": "Prescriptive",

"orga\_name": "Entity Pharmaceuticals Ltd",

"task\_activity": "Ensure that the consolidation of financial statements of the company shall be made in accordance with the provisions of Schedule III of the Act and the applicable accounting standards",

"task\_legi\_name": "The Companies Act, 2013",

"task\_cat\_law\_id": 2,

"loca\_id": 34,

"task\_cat\_law": "Corporate",

"tmap\_client\_task\_id": "D34240000722",

"task\_sub\_event": "Preparation of Financial Statements",

"loca\_name": "Emcure Pharma- Secreterial",

"task\_rule\_id": 829,

"task\_pr\_due\_date": "21-05-2021",

"task\_event": "Accounts of Companies",

"task\_procedure": "NA",

"task\_type\_of\_task": "Verification",

"task\_legi\_id": 135,

"task\_rule\_name": "The Companies (Accounts) Rules, 2014",

"task\_impact": "Major",

"task\_frequency\_for\_operation": "Yearly",

"function\_head\_name": "Jayant Prakash",

"orga\_id": 2,

"dept\_name": "Secretarial",

"performer\_name": "Sheetal Kulkarni",

"task\_activity\_when": "At the time of consolidation of accounts",

"task\_activity\_who": "Company having one or more subsidiaries or associate companies",

"task\_reference": "Second Proviso to Section 129(3) and Rule 6",

"ttrn\_status": "Active",

"performer\_id": 169,

"reviewer\_name": "Jayant Prakash",

"document\_attached": 0,

"reviewer\_id": 316,

"dept\_id": 24,

"task\_legal\_due\_date": "29-05-2021",

"task\_frequency": "Yearly"

}

],

"TotalNoOfTasks": 204

},

"status": "Success"

}

}

Description: get list of upcoming task

Function : TasksController.listOfUpcommingTask()

* PATH: /waitingforapprovaltasks Method: POST

Input: <http://localhost:9002/compliance/waitingforapprovaltasks>

Response: JSON String

{

"response": {

"data": {

"TasksFilter": [

{

"Legislation": [

{

"task\_legi\_id": 34,

"loca\_id": 3,

"orga\_id": 2,

"task\_legi\_name": "The Maternity Benefit Act, 1961 ",

"dept\_id": 3,

"task\_cat\_law\_id": 5

},

{

"task\_legi\_id": 203,

"loca\_id": 3,

"orga\_id": 2,

"task\_legi\_name": "The Sexual Harassment of Women at Workplace (Prevention, Prohibition and Redressal) Act, 2013 ",

"dept\_id": 3,

"task\_cat\_law\_id": 5

}

],

"Sub\_Event": [

{

"task\_legi\_id": 34,

"loca\_id": 3,

"task\_Sub\_Event": "NA",

"task\_Event": "NA",

"orga\_id": 2,

"task\_rule\_id": 42,

"dept\_id": 3,

"task\_cat\_law\_id": 5

},

{

"task\_legi\_id": 203,

"loca\_id": 3,

"task\_Sub\_Event": "NA",

"task\_Event": "NA",

"orga\_id": 2,

"task\_rule\_id": 138,

"dept\_id": 3,

"task\_cat\_law\_id": 5

}

],

"Type\_Of\_Task": [

{

"task\_type\_of\_task": "Returns"

},

{

"task\_type\_of\_task": "Register and Records"

}

],

"Event": [

{

"task\_legi\_id": 34,

"loca\_id": 3,

"task\_Event": "NA",

"orga\_id": 2,

"task\_rule\_id": 42,

"dept\_id": 3,

"task\_cat\_law\_id": 5

},

{

"task\_legi\_id": 203,

"loca\_id": 3,

"task\_Event": "NA",

"orga\_id": 2,

"task\_rule\_id": 138,

"dept\_id": 3,

"task\_cat\_law\_id": 5

}

],

"Frequency": [

{

"task\_frequency": "Yearly"

}

],

"Rule": [

{

"task\_legi\_id": 34,

"task\_rule\_name": "The Maharashtra Maternity Benefit Rules, 1965 ",

"task\_rule\_id": 42,

"task\_cat\_law\_id": 5

},

{

"task\_legi\_id": 203,

"task\_rule\_name": "The Sexual Harassment of Women at Workplace (Prevention, Prohibition and Redressal) Rules, 2013",

"task\_rule\_id": 138,

"task\_cat\_law\_id": 5

}

],

"Cat\_law": [

{

"task\_cat\_law": "Labour Laws",

"task\_cat\_law\_id": 5

}

]

}

],

"OrganogramFilter": [

{

"Entity": [

{

"orga\_name": "Entity Pharmaceuticals Ltd",

"orga\_id": 2

}

],

"Function": [

{

"loca\_id": 3,

"orga\_id": 2,

"dept\_name": "HR",

"dept\_id": 3

}

],

"Executor": [

{

"loca\_id": 3,

"user\_id": 3,

"user\_name": "Deepak Patil",

"orga\_id": 2,

"dept\_id": 3

}

],

"Unit": [

{

"loca\_id": 3,

"loca\_name": "Emcure Warehouse Mann",

"orga\_id": 2

}

],

"Evaluator": [

{

"loca\_id": 3,

"user\_id": 148,

"user\_name": "Prakash Khandare",

"orga\_id": 2,

"dept\_id": 3

}

]

}

],

"AllTasks": [

{

"task\_prohibitive": "Prescriptive",

"orga\_name": "Entity Pharmaceuticals Ltd",

"task\_activity": "File return with the Competent Authority in Form 11 before the 15th day of January every year",

"task\_legi\_name": "The Maternity Benefit Act, 1961 ",

"task\_cat\_law\_id": 5,

"loca\_id": 3,

"task\_cat\_law": "Labour Laws",

"tmap\_client\_task\_id": "D03030003311",

"task\_sub\_event": "NA",

"loca\_name": "Emcure Warehouse Mann",

"task\_rule\_id": 42,

"task\_pr\_due\_date": "06-01-2020",

"task\_event": "NA",

"task\_procedure": "NA",

"task\_type\_of\_task": "Returns",

"task\_legi\_id": 34,

"task\_rule\_name": "The Maharashtra Maternity Benefit Rules, 1965 ",

"task\_impact": "Major",

"task\_frequency\_for\_operation": "Yearly",

"function\_head\_name": "Gautam Sable",

"orga\_id": 2,

"dept\_name": "HR",

"performer\_name": "Deepak Patil",

"task\_activity\_when": "On applicability of the Act",

"task\_activity\_who": "Employer",

"task\_reference": "Rule 15",

"ttrn\_status": "Partially\_Completed",

"performer\_id": 3,

"reviewer\_name": "Prakash Khandare",

"document\_attached": 0,

"reviewer\_id": 148,

"dept\_id": 3,

"task\_legal\_due\_date": "14-01-2020",

"task\_frequency": "Yearly"

},

{

"task\_prohibitive": "Prescriptive",

"orga\_name": "Entity Pharmaceuticals Ltd",

"task\_activity": "File return with the Competent Authority in Form 11 before the 15th day of January every year",

"task\_legi\_name": "The Maternity Benefit Act, 1961 ",

"task\_cat\_law\_id": 5,

"loca\_id": 3,

"task\_cat\_law": "Labour Laws",

"tmap\_client\_task\_id": "D03030003311",

"task\_sub\_event": "NA",

"loca\_name": "Emcure Warehouse Mann",

"task\_rule\_id": 42,

"task\_pr\_due\_date": "06-01-2021",

"task\_event": "NA",

"task\_procedure": "NA",

"task\_type\_of\_task": "Returns",

"task\_legi\_id": 34,

"task\_rule\_name": "The Maharashtra Maternity Benefit Rules, 1965 ",

"task\_impact": "Major",

"task\_frequency\_for\_operation": "Yearly",

"function\_head\_name": "Gautam Sable",

"orga\_id": 2,

"dept\_name": "HR",

"performer\_name": "Deepak Patil",

"task\_activity\_when": "On applicability of the Act",

"task\_activity\_who": "Employer",

"task\_reference": "Rule 15",

"ttrn\_status": "Partially\_Completed",

"performer\_id": 3,

"reviewer\_name": "Prakash Khandare",

"document\_attached": 0,

"reviewer\_id": 148,

"dept\_id": 3,

"task\_legal\_due\_date": "14-01-2021",

"task\_frequency": "Yearly"

}

],

"TotalNoOfTasks": 2

},

"status": "Success"

}

}

Description: get list of tasks which are in Waiting For Approval state

Function : TasksController.listWaitingForApprovalTasksTask()

* PATH: /reopenedtasks Method: POST

Input: <http://localhost:9002/compliance/reopenedtasks>

Response: JSON String

{

"response": {

"data": {

"TasksFilter": [

{

"Legislation": [

{

"task\_legi\_id": 192,

"loca\_id": 1,

"orga\_id": 2,

"task\_legi\_name": "The Employment Exchange (Compulsory Notification of Vacancies) Act, 1959",

"dept\_id": 3,

"task\_cat\_law\_id": 5

},

{

"task\_legi\_id": 30,

"loca\_id": 1,

"orga\_id": 2,

"task\_legi\_name": "The Industrial Disputes Act, 1947 ",

"dept\_id": 3,

"task\_cat\_law\_id": 5

},

{

"task\_legi\_id": 22,

"loca\_id": 1,

"orga\_id": 2,

"task\_legi\_name": "The Payment of Wages Act, 1936",

"dept\_id": 3,

"task\_cat\_law\_id": 5

}

],

"Sub\_Event": [

{

"task\_legi\_id": 192,

"loca\_id": 1,

"task\_Sub\_Event": "NA",

"task\_Event": "NA",

"orga\_id": 2,

"task\_rule\_id": 104,

"dept\_id": 3,

"task\_cat\_law\_id": 5

},

{

"task\_legi\_id": 30,

"loca\_id": 1,

"task\_Sub\_Event": "NA",

"task\_Event": "NA",

"orga\_id": 2,

"task\_rule\_id": 36,

"dept\_id": 3,

"task\_cat\_law\_id": 5

},

{

"task\_legi\_id": 22,

"loca\_id": 1,

"task\_Sub\_Event": "NA",

"task\_Event": "NA",

"orga\_id": 2,

"task\_rule\_id": 30,

"dept\_id": 3,

"task\_cat\_law\_id": 5

}

],

"Type\_Of\_Task": [

{

"task\_type\_of\_task": "Submission"

},

{

"task\_type\_of\_task": "Notices"

},

{

"task\_type\_of\_task": "Intimation"

},

{

"task\_type\_of\_task": "Payment"

}

],

"Event": [

{

"task\_legi\_id": 192,

"loca\_id": 1,

"task\_Event": "NA",

"orga\_id": 2,

"task\_rule\_id": 104,

"dept\_id": 3,

"task\_cat\_law\_id": 5

},

{

"task\_legi\_id": 30,

"loca\_id": 1,

"task\_Event": "NA",

"orga\_id": 2,

"task\_rule\_id": 36,

"dept\_id": 3,

"task\_cat\_law\_id": 5

},

{

"task\_legi\_id": 22,

"loca\_id": 1,

"task\_Event": "NA",

"orga\_id": 2,

"task\_rule\_id": 30,

"dept\_id": 3,

"task\_cat\_law\_id": 5

}

],

"Frequency": [

{

"task\_frequency": "Quarterly"

},

{

"task\_frequency": "Half\_Yearly"

}

],

"Rule": [

{

"task\_legi\_id": 192,

"task\_rule\_name": "The Employment Exchange (Compulsory Notification of Vacancies) Rules, 1960 ",

"task\_rule\_id": 104,

"task\_cat\_law\_id": 5

},

{

"task\_legi\_id": 30,

"task\_rule\_name": "The Industrial Disputes (Bombay) Rules, 1957",

"task\_rule\_id": 36,

"task\_cat\_law\_id": 5

},

{

"task\_legi\_id": 22,

"task\_rule\_name": "The Maharashtra Payment Of Wages Rules, 1963",

"task\_rule\_id": 30,

"task\_cat\_law\_id": 5

}

],

"Cat\_law": [

{

"task\_cat\_law": "Labour Laws",

"task\_cat\_law\_id": 5

}

]

}

],

"OrganogramFilter": [

{

"Entity": [

{

"orga\_name": "Entity Pharmaceuticals Ltd",

"orga\_id": 2

}

],

"Function": [

{

"loca\_id": 1,

"orga\_id": 2,

"dept\_name": "HR",

"dept\_id": 3

}

],

"Executor": [

{

"loca\_id": 1,

"user\_id": 351,

"user\_name": "Nikesh Khaire",

"orga\_id": 2,

"dept\_id": 3

},

{

"loca\_id": 1,

"user\_id": 153,

"user\_name": "Ulhas Divekar",

"orga\_id": 2,

"dept\_id": 3

}

],

"Unit": [

{

"loca\_id": 1,

"loca\_name": "Hinjewadi EOU",

"orga\_id": 2

}

],

"Evaluator": [

{

"loca\_id": 1,

"user\_id": 148,

"user\_name": "Prakash Khandare",

"orga\_id": 2,

"dept\_id": 3

},

{

"loca\_id": 1,

"user\_id": 167,

"user\_name": "Pravin Bhosale",

"orga\_id": 2,

"dept\_id": 3

}

]

}

],

"AllTasks": [

{

"task\_prohibitive": "Prescriptive",

"orga\_name": "Entity Pharmaceuticals Ltd",

"task\_activity": "Furnish quarterly returns in Form ER I to the Local Employment Exchange",

"task\_legi\_name": "The Employment Exchange (Compulsory Notification of Vacancies) Act, 1959",

"task\_cat\_law\_id": 5,

"loca\_id": 1,

"task\_cat\_law": "Labour Laws",

"tmap\_client\_task\_id": "D01030004671",

"task\_sub\_event": "NA",

"loca\_name": "Hinjewadi EOU",

"task\_rule\_id": 104,

"task\_pr\_due\_date": "21-04-2020",

"task\_event": "NA",

"task\_procedure": "NA",

"task\_type\_of\_task": "Submission",

"task\_legi\_id": 192,

"task\_rule\_name": "The Employment Exchange (Compulsory Notification of Vacancies) Rules, 1960 ",

"task\_impact": "Low",

"task\_frequency\_for\_operation": "Quarterly",

"function\_head\_name": "Gautam Sable",

"orga\_id": 2,

"dept\_name": "HR",

"performer\_name": "Nikesh Khaire",

"task\_activity\_when": "On applicability of the Act",

"task\_activity\_who": "Employer",

"task\_reference": "Section 5 (2) and Rule 6",

"ttrn\_status": "Re\_Opened",

"performer\_id": 351,

"reviewer\_name": "Prakash Khandare",

"document\_attached": 0,

"reviewer\_id": 148,

"dept\_id": 3,

"task\_legal\_due\_date": "29-04-2020",

"task\_frequency": "Quarterly"

}

],

"TotalNoOfTasks": 1

},

"status": "Success"

}

}

Description: get list of reopened tasks Function : TasksController.reopenedtasks(

* PATH: /searchrepository Method: POST

Input: <http://localhost:9002/compliance/searchrepository>

{

"orga\_id":0,

"loca\_id":0,

"dept\_id":0,

"executor\_id":0,

"evaluator\_id":0,

"cat\_law\_id":0,

"impact":"null",

"proh\_pres":"null",

"type\_of\_task":"null",

"frequency":"null",

"task\_status":"null",

"event":"null",

"sub\_event":"null",

"legi\_id":0,

"rule\_id":0,

"task\_id":"null",

"nextSearch":0,

"itemsPerPage":10,

"pageno":1

}

Response: JSON String

{

"response": {

"data": {

"repoData": [

{

"tmap\_client\_task\_id": "D05030002838",

"task\_legi\_name": "The Child and Adolescent Labour (Prohibition and Regulation) Act, 1986",

"task\_rule\_name": "The Child Labour (Prohibition and Regulation) Rules, 1988",

"task\_pr\_due\_date": "25-04-2021",

"task\_legal\_due\_date": "30-04-2021",

"task\_activity\_who": "Occupier",

"task\_activity\_when": "On applicability of the act",

"task\_activity": "Ensure that no child is allowed or permitted to work in any of the occupations set fourth in Part A of the Schedule or in any workshop wherein any of the processes set forth in Part B of the Schedule is carried on",

"task\_procedure": "NA",

"task\_impact": "Major",

"task\_frequency\_for\_operation": "Half\_Yearly",

"task\_reference": "Section 3",

"task\_cat\_law": "Labour Laws",

"task\_type\_of\_task": "Welfare",

"task\_prohibitive": "Prohibitive",

"task\_event": "NA",

"task\_sub\_event": "NA",

"ttrn\_status": "Active",

"orga\_id": "2",

"orga\_name": "Entity Pharmaceuticals Ltd",

"loca\_id": "5",

"loca\_name": "Hinjewadi SPD",

"dept\_id": "3",

"dept\_name": "HR",

"performer\_id": "3",

"performer\_name": "Deepak Patil",

"reviewer\_id": "148",

"reviewer\_name": "Prakash Khandare",

"task\_cat\_law\_id": "5",

"task\_legi\_id": "29",

"task\_rule\_id": "153",

"function\_head\_name": "Ashish Bhide",

"document\_attached": 0

},

{

"tmap\_client\_task\_id": "D05030002839",

"task\_legi\_name": "The Child and Adolescent Labour (Prohibition and Regulation) Act, 1986",

"task\_rule\_name": "The Child Labour (Prohibition and Regulation) Rules, 1988",

"task\_pr\_due\_date": "27-12-2020",

"task\_legal\_due\_date": "31-12-2020",

"task\_activity\_who": "Occupier",

"task\_activity\_when": "On applicability of the act",

"task\_activity": "Display in a conspicuous and accessible place at the place of work a notice in the local language and in the English language containing an abstract of sections 3 and 14",

"task\_procedure": "NA",

"task\_impact": "Major",

"task\_frequency\_for\_operation": "Yearly",

"task\_reference": "Section 12",

"task\_cat\_law": "Labour Laws",

"task\_type\_of\_task": "Display",

"task\_prohibitive": "Prescriptive",

"task\_event": "NA",

"task\_sub\_event": "NA",

"ttrn\_status": "Active",

"orga\_id": "2",

"orga\_name": "Entity Pharmaceuticals Ltd",

"loca\_id": "5",

"loca\_name": "Hinjewadi SPD",

"dept\_id": "3",

"dept\_name": "HR",

"performer\_id": "3",

"performer\_name": "Deepak Patil",

"reviewer\_id": "148",

"reviewer\_name": "Prakash Khandare",

"task\_cat\_law\_id": "5",

"task\_legi\_id": "29",

"task\_rule\_id": "153",

"function\_head\_name": "Ashish Bhide",

"document\_attached": 0

}

]

},

"status": "Success"

}

}

Description: search repostory and returns the results Function : TasksController.searchRepository()

* PATH: /scrollrepository Method: POST

Input: <http://localhost:9002/compliance/searchrepository>

{

"orga\_id":0,

"loca\_id":0,

"dept\_id":0,

"executor\_id":0,

"evaluator\_id":0,

"cat\_law\_id":0,

"impact":"null",

"proh\_pres":"null",

"type\_of\_task":"null",

"frequency":"null",

"task\_status":"null",

"event":"null",

"sub\_event":"null",

"legi\_id":0,

"rule\_id":0,

"task\_id":"null",

"nextSearch":0,

"itemsPerPage":10,

"pageno":1

}

Response: JSON String

{

"response": {

"data": {

"repoData": [

{

"tmap\_client\_task\_id": "D05030002838",

"task\_legi\_name": "The Child and Adolescent Labour (Prohibition and Regulation) Act, 1986",

"task\_rule\_name": "The Child Labour (Prohibition and Regulation) Rules, 1988",

"task\_pr\_due\_date": "25-04-2021",

"task\_legal\_due\_date": "30-04-2021",

"task\_activity\_who": "Occupier",

"task\_activity\_when": "On applicability of the act",

"task\_activity": "Ensure that no child is allowed or permitted to work in any of the occupations set fourth in Part A of the Schedule or in any workshop wherein any of the processes set forth in Part B of the Schedule is carried on",

"task\_procedure": "NA",

"task\_impact": "Major",

"task\_frequency\_for\_operation": "Half\_Yearly",

"task\_reference": "Section 3",

"task\_cat\_law": "Labour Laws",

"task\_type\_of\_task": "Welfare",

"task\_prohibitive": "Prohibitive",

"task\_event": "NA",

"task\_sub\_event": "NA",

"ttrn\_status": "Active",

"orga\_id": "2",

"orga\_name": "Entity Pharmaceuticals Ltd",

"loca\_id": "5",

"loca\_name": "Hinjewadi SPD",

"dept\_id": "3",

"dept\_name": "HR",

"performer\_id": "3",

"performer\_name": "Deepak Patil",

"reviewer\_id": "148",

"reviewer\_name": "Prakash Khandare",

"task\_cat\_law\_id": "5",

"task\_legi\_id": "29",

"task\_rule\_id": "153",

"function\_head\_name": "Ashish Bhide",

"document\_attached": 0

},

{

"tmap\_client\_task\_id": "D05030002839",

"task\_legi\_name": "The Child and Adolescent Labour (Prohibition and Regulation) Act, 1986",

"task\_rule\_name": "The Child Labour (Prohibition and Regulation) Rules, 1988",

"task\_pr\_due\_date": "27-12-2020",

"task\_legal\_due\_date": "31-12-2020",

"task\_activity\_who": "Occupier",

"task\_activity\_when": "On applicability of the act",

"task\_activity": "Display in a conspicuous and accessible place at the place of work a notice in the local language and in the English language containing an abstract of sections 3 and 14",

"task\_procedure": "NA",

"task\_impact": "Major",

"task\_frequency\_for\_operation": "Yearly",

"task\_reference": "Section 12",

"task\_cat\_law": "Labour Laws",

"task\_type\_of\_task": "Display",

"task\_prohibitive": "Prescriptive",

"task\_event": "NA",

"task\_sub\_event": "NA",

"ttrn\_status": "Active",

"orga\_id": "2",

"orga\_name": "Entity Pharmaceuticals Ltd",

"loca\_id": "5",

"loca\_name": "Hinjewadi SPD",

"dept\_id": "3",

"dept\_name": "HR",

"performer\_id": "3",

"performer\_name": "Deepak Patil",

"reviewer\_id": "148",

"reviewer\_name": "Prakash Khandare",

"task\_cat\_law\_id": "5",

"task\_legi\_id": "29",

"task\_rule\_id": "153",

"function\_head\_name": "Ashish Bhide",

"document\_attached": 0

}

]

},

"status": "Success"

}

}

Description: get repository when scrolling Function : TasksController.getRepositoryOnScroll()

* PATH: /paginationEx/get Method: GET

Response: JSON String Description: for pagination

Function : TasksController.paginationExample()

Status: Function Not Available

* PATH: /getalltasks Method: POST

Input: <http://localhost:9002/compliance/getalltasks>

Response: JSON String

{

"response": {

"data": {

"TasksFilter": [

{

"Legislation": [

{

"task\_legi\_id": 137,

"loca\_id": 1,

"orga\_id": 2,

"task\_legi\_name": "The Environment (Protection) Act,1986",

"dept\_id": 7

},

{

"task\_legi\_id": 712,

"loca\_id": 1,

"orga\_id": 2,

"task\_legi\_name": "The Central Goods and Services Tax Act, 2017",

"dept\_id": 1

}

],

"Sub\_Event": [

{

"task\_legi\_id": 137,

"loca\_id": 1,

"task\_Sub\_Event": "NA",

"task\_Event": "NA",

"orga\_id": 2,

"task\_rule\_id": 293,

"dept\_id": 7,

"task\_cat\_law\_id": 3

},

{

"task\_legi\_id": 712,

"loca\_id": 1,

"task\_Sub\_Event": "NA",

"task\_Event": "NA",

"orga\_id": 2,

"task\_rule\_id": 1376,

"dept\_id": 1,

"task\_cat\_law\_id": 8

}

],

"Event": [

{

"task\_legi\_id": 137,

"loca\_id": 1,

"task\_Event": "NA",

"orga\_id": 2,

"task\_rule\_id": 293,

"dept\_id": 7,

"task\_cat\_law\_id": 3

},

{

"task\_legi\_id": 712,

"loca\_id": 1,

"task\_Event": "NA",

"orga\_id": 2,

"task\_rule\_id": 1376,

"dept\_id": 1,

"task\_cat\_law\_id": 8

}

],

"Rule": [

{

"task\_legi\_id": 137,

"task\_rule\_name": "The Environment (Protection) Rules, 1986 ",

"task\_rule\_id": 293

},

{

"task\_legi\_id": 712,

"task\_rule\_name": "The Central Goods and Services Tax Rules, 2017",

"task\_rule\_id": 1376

}

]

}

],

"OrganogramFilter": [

{

"Executor": [

{

"loca\_id": 1,

"user\_id": 7,

"user\_name": "Ajit Nagtilak",

"orga\_id": 2

},

{

"loca\_id": 1,

"user\_id": 325,

"user\_name": "Nitin Salunkhe",

"orga\_id": 2

}

],

"FunctionHead": [

{

"loca\_id": 1,

"user\_id": 91,

"user\_name": "Bharat Dighe",

"orga\_id": 2

},

{

"loca\_id": 1,

"user\_id": 317,

"user\_name": "Nitesh Akarte",

"orga\_id": 2

}

],

"Evaluator": [

{

"loca\_id": 1,

"user\_id": 42,

"user\_name": "Prashant Rane",

"orga\_id": 2

},

{

"loca\_id": 1,

"user\_id": 47,

"user\_name": "Krishna Kadam",

"orga\_id": 2

}

]

}

]

},

"status": "Success"

}

}

Description: get all repository tasks Function : TasksController.getalltasks()

* PATH: /getCategory Method: GET

Input: <http://localhost:9002/compliance/getCategory>

Response: JSON String

{

"response": {

"data": [

{

"cat\_law\_name": "Regulatory",

"cat\_law\_id": 7

},

{

"cat\_law\_name": "Labour Laws",

"cat\_law\_id": 5

},

{

"cat\_law\_name": "Commercial",

"cat\_law\_id": 1

},

{

"cat\_law\_name": "Environment Laws",

"cat\_law\_id": 3

},

{

"cat\_law\_name": "Taxation- Direct & Indirect",

"cat\_law\_id": 8

},

{

"cat\_law\_name": "Corporate",

"cat\_law\_id": 2

},

{

"cat\_law\_name": "Internal",

"cat\_law\_id": 16

},

{

"cat\_law\_name": "Information Technology",

"cat\_law\_id": 13

},

{

"cat\_law\_name": "Repealed",

"cat\_law\_id": 17

},

{

"cat\_law\_name": "Local",

"cat\_law\_id": 6

}

],

"status": "Success"

}

}

Description: get the category

Function : TasksController.getCategory()

* PATH: /getTypeOfTask Method: GET

Input: <http://localhost:9002/compliance/getTypeOfTask>

Response: JSON String

{

"response": {

"data": [

{

"task\_type\_of\_task": "Verification"

},

{

"task\_type\_of\_task": "Registration"

},

,

{

"task\_type\_of\_task": "Intimation"

},

{

"task\_type\_of\_task": "Reports"

},

{

"task\_type\_of\_task": "Committee"

},

{

"task\_type\_of\_task": "Meetings"

},

{

"task\_type\_of\_task": "Transfer or Conversion"

},

{

"task\_type\_of\_task": "Employment and Nomination"

},

{

"task\_type\_of\_task": "Insurance"

}

],

"status": "Success"

}

}

Description: get tasks by type

Function : TasksController.getTypeOfTask()

* PATH: /getFrequencyList Method: POST

Input: <http://localhost:9002/compliance/getFrequencyList>

Response: JSON String

{

"response": {

"data": [

{

"task\_frequency": "One\_Time"

},

{

"task\_frequency": "Yearly"

},

{

"task\_frequency": "Ongoing"

},

{

"task\_frequency": "Event\_Based"

},

{

"task\_frequency": "Monthly"

},

{

"task\_frequency": "Five\_Yearly"

},

{

"task\_frequency": "Half\_Yearly"

},

{

"task\_frequency": "Quarterly"

},

{

"task\_frequency": "Two\_Yearly"

},

{

"task\_frequency": "Four\_Yearly"

},

{

"task\_frequency": "Three\_Yearly"

},

{

"task\_frequency": "Fourteen\_Monthly"

},

{

"task\_frequency": "Seven\_Yearly"

},

{

"task\_frequency": "Weekly"

},

{

"task\_frequency": "Four\_Monthly"

}

],

"status": "Success"

}

}

Description: get frequency list

Function : TasksController.getFrequencyList()

* PATH: /savetasksconfiguration Method: POST

Input:

<http://localhost:9002/compliance/savetasksconfiguration>

{

 "ttrn\_legal\_due\_date":"31-05-2021",

 "ttrn\_uh\_due\_date":"31-05-2021",

 "ttrn\_fh\_due\_date":"31-05-2021",

 "ttrn\_rw\_due\_date":"31-05-2021",

 "ttrn\_pr\_due\_date":"31-05-2021",

 "ttrn\_impact\_on\_organization":"Major",

 "ttrn\_impact\_on\_unit":"Severe",

 "ttrn\_impact":"Severe",

 "ttrn\_prior\_days\_buffer":0,

 "ttrn\_alert\_days":0,

 "ttrn\_document":"0",

 "ttrn\_historical":"0",

 "ttrn\_allow\_back\_date\_completion":0,

 "ttrn\_allow\_approver\_reopening":0,

 "ttrn\_no\_of\_back\_days\_allowed":0,

 "tasks\_list":[

         {"ttrn\_client\_task\_id":"D34240001194",

          "ttrn\_performer\_user\_id":355}

        ],

 "ttrn\_frequency\_for\_operation":"User\_Defined",

 "validate\_dates":"TRUE",

 "ttrn\_first\_alert":"",

 "ttrn\_second\_alert":"",

 "ttrn\_third\_alert":"",

 "ttrn\_frequency\_for\_alerts":"User\_Defined"

}

Response: JSON String

{

    "response": {

        "data": [

            {

                "successClientTasksId": "D34240001194"

            }

        ],

        "status": "Success"

    }

}

Description: save task configuration details Function :

TasksConfigurationController.savetasksconfiguration()

* PATH: / deleteTaskDocument

Method: POST

Input: <http://localhost:9002/compliance/deleteTaskDocument>

{

"udoc\_id":761

}

Response: JSON String

{

"response": {

"message": "Documet deleted successfully.",

"status": "Success"

}

}

Description: save task configuration details Function :

TasksConfigurationController. deleteTaskDocument()

* PATH: /updatetasksconfiguration Method: POST

Input: <http://localhost:9002/compliance/updatetasksconfiguration>

{

"validate\_dates":"TRUE",

"ttrn\_prior\_days\_buffer":2,

"ttrn\_id":934,

"ttrn\_legal\_due\_date":"25-04-2021",

"ttrn\_uh\_due\_date":"23-04-2021",

"ttrn\_fh\_due\_date":"21-04-2021",

"ttrn\_rw\_due\_date":"19-04-2021",

"ttrn\_pr\_due\_date":"17-04-2021",

"ttrn\_performer\_name":"Amruta Boodle",

"user\_email":"Amruta.Boodle@emcure.co.in"

}

Response: JSON String

{

"response": {

"status": "Success"

}

}

Description: update the task configuration details Function :

TasksConfigurationController.updatetasksconfiguration()

* PATH: /activationoftasks Method: POST

Input: <http://localhost:9002/compliance/activationoftasks>

{

"operation\_to\_perform" : "activate",

"tasks\_list": [{

"ttrn\_id":934

}]

}

Response: JSON String

{

"response": {

"status": "Success"

}

}

Description: activate or deactivate the task configuration Function : TasksConfigurationController.activationOfTasks()

* PATH: /getallmappedtasksforconfigurationpage Method: POST

Input: /compliance/getallmappedtasksforconfigurationpage

{

"searching\_for":"tasksconfiguration",

"country\_id":0,

"state\_id":0,

"cat\_id": 0,

"orga\_id":0,

"legi\_id":0,

"rule\_id":0,

"rule\_id":0,

"dept\_id":0,

"executor":0,

"evaluator":0,

"frequency":"NA",

"loca\_list":[]

}

Response: JSON String

{

"response": {

"data": [

{

"task\_rule\_name": "The Companies (Meetings of Board and its Powers) Rules, 2014\n",

"task\_impact": "Major",

"task\_impact\_on\_orga": "Major",

"task\_frequency\_for\_operation": "Event\_Based",

"task\_activity": "Take prior approval by means of passing a special resolution at a general meeting",

"task\_impact\_on\_unit": "Major",

"task\_legi\_name": "The Companies Act, 2013",

"task\_activity\_when": "In case the aggregate of the loans and investment so far made, the amount for which guarantee or security so far provided to or in all other bodies corporate along with the investment, loan, guarantee or security proposed to be made or given by the Board, exceed the limits specified under section 186(2)",

"exec\_id": 45,

"task\_activity\_who": "Company",

"task\_reference": "Section 186(3) and Rule 11, 13",

"tmap\_client\_task\_id": "D34240000675",

"task\_lexcare\_id": "C0006254"

}

],

"status": "Success"

}

}

Description: get all the tasks which are mapped for configuration page

Function : TasksConfigurationController.getAllMappedTasksForConfigurationPage()

* PATH: /savetaskcompletion Method: POST

Input: <http://localhost:9002/compliance/savetaskcompletion>

{

"ttrn\_completed\_date":"30-03-2021",

"ttrn\_performer\_comments":"Financial Statements will be prepared and finalized between June-August 2021",

"ttrn\_reason\_for\_non\_compliance":"",

"ttrn\_ids":[{"ttrn\_id":40200}],

"ttrn\_event\_not\_occure":"No"

}

Response: JSON String

{

    "response": {

        "message": "Saved Successfully.",

        "status": "Success"

    }

}

Description:update the task as a complete and send an email for executor on completion of task

Function : TasksConfigurationController.savetaskcompletion()

* PATH: /searchTaskForDefaultConfiguration Method: POST

Response: JSON String

{

     "response": {

         "data": {

             "search\_list": []

         },

         "status": "Success"

     }

}

Description: search clinical task

Function :

TasksConfigurationController.searchClinicalTaskForConfiguration()

* PATH: /saveDefaultTaskConfiguration Method: POST

Input : <http://localhost:9002/compliance/saveDefaultTaskConfiguration>

{

    "dtco\_after\_before":"Same",

    "dtco\_event":"NO",

    "dtco\_sub\_event":"NO",

    "dtco\_legal\_days" :10,

    "dtco\_uh\_days":10,

    "dtco\_fh\_days":10,

    "dtco\_rw\_days":10,

    "dtco\_pr\_days":10,

    "tasks\_list":[{"dtco\_client\_task\_id":1}

        ]

}

Response: JSON String

{

    "response": {

        "dtco\_uh\_days": 10,

        "dtco\_sub\_event": "NO",

        "dtco\_fh\_days": 10,

        "data": {},

        "dtco\_pr\_days": 10,

        "dtco\_rw\_days": 10,

        "dtco\_after\_before": "Same",

        "dtco\_event": "NO",

        "dtco\_legal\_days": 10,

        "tasks\_list": [

            {

                "dtco\_client\_task\_id": 1

            }

        ],

        "status": "Success"

    }

}

Description: save clinical task configuration

Function :

TasksConfigurationController.saveClinicalConfiguration()

* PATH: /initiateDefaultConfiguredTask Method: POST

Input: <http://localhost:9002/compliance/initiateDefaultConfiguredTask>

{

    "date\_of\_initiation":"06-05-2021",

    "comments": "Noo",

    "tasks\_list":[{

        "dtco\_id": 1

    }]

}

Response: JSON String

{

    "response": {

        "comments": "Noo",

        "date\_of\_initiation": "06-05-2021",

        "tasks\_list": [

            {

                "dtco\_id": 1

            }

        ],

        "status": "Success"

    }

}

Description: initiate the default configured task

Function :

TasksConfigurationController.initiateDefaultConfiguredTask()

* PATH: /approveTask Method: POST

Input: <http://localhost:9002/compliance/approveTask>

{

"ttrn\_id":12

}

Response: JSON String

{

"response": {

"status": "Success"

}

}

Description: approve task

Function : TasksConfigurationController.approveTask()

* PATH: /reopenTask Method: POST

Input: <http://localhost:9002/compliance/reopenTask>

{

"ttrn\_id":12,

"reopen\_comment":"NO Document attached"

}

Response: JSON String

{

"response": {

"status": "Success"

}

}

Description: reopen the task

Function : TasksConfigurationController.reopenTask()

* PATH: /deleteTaskHistory Method: POST

Input: <http://localhost:9002/compliance/deleteTaskHistory>

{

"ttrn\_completed\_date":"2021-05-26T07:04:07.035Z",

"ttrn\_id":103,

"ttrn\_client\_task\_id":"D35240001454"

}

Response: JSON String

{

"response": {

"message": "Task deleted successfully",

"status": "Success"

}

}

Description: delete task history

Function : TasksConfigurationController.deleteTaskHistory()

* PATH: /getDefaultTaskList Method: POST

Input : <http://localhost:9002/compliance/getDefaultTaskList>

Response: JSON String

{

    "response": {

        "data": [

            {

                "dtco\_uh\_days": 0,

                "tmap\_dept\_id": 3,

                "dtco\_pr\_days": 0,

                "dtco\_rw\_days": 0,

                "dtco\_event": "Audit and Auditors",

                "dtco\_legal\_days": 0,

                "dtco\_frequency": "Event\_Based",

                "dtco\_client\_task\_id": "D27050000540",

                "tmap\_orga\_id": 2,

                "tmap\_rw\_id": 1,

                "tmap\_pr\_id": 1,

                "dtco\_fh\_days": 0,

                "dtco\_sub\_event": "Audit conducted by Auditor",

                "dtco\_id": 1,

                "dtco\_after\_before": "Same",

                "tmap\_loca\_id": 2,

                "lexcare\_task\_id": "L0005670"

            }

        ],

        "status": "Success"

    }

}

Description: get the default task list

Function : TasksConfigurationController.getDefaultTaskList()

* PATH: /updateDefaultTaskConfiguration Method: POST

Input: <http://localhost:9002/compliance/updateDefaultTaskConfiguration>

{

    "dtco\_id":"1",

    "dtco\_client\_task\_id":"D27050000541",

    "dtco\_after\_before":"Same",

    "dtco\_pr\_days":"10",

    "dtco\_rw\_days" :"10",

    "dtco\_fh\_days":"10",

    "dtco\_uh\_days":"10",

    "dtco\_legal\_days":"10"

}

Response: JSON String

{

    "response": {

        "status": "Success"

    }

}

Description: update default task configuration

Function :

TasksConfigurationController.updateDefaultTaskConfiguration()

* PATH: /updateTaskReasonForNonCompliance Method: POST

Input: /compliance/updateTaskReasonForNonCompliance

{

"ttrn\_id": "1",

"ttrn\_reason\_for\_non\_compliance":"No reason"

}

Response: JSON String

{

"response": {

"message": "Reason of non-compliance updated successfully.",

"status": "Success"

}

}

Description: update task reason for non complaince

Function :

TasksConfigurationController.updateTaskReasonForNonCompliance()

* PATH: /savesubtaskcompletion Method: POST

Response: JSON String

Description: save subtask completion

Function : TasksConfigurationController.savesubtaskcompletion()

Status: Function Not Available

* PATH: /searchConfiguartionPage Method: POST

Input: <http://127.0.0.1:8082/searchConfiguartionPage>

{

"country\_id": 1,

"state\_id":2,

"cat\_id": 7,

"legi\_id": 534,

"rule\_id":1067,

"orga\_id": 0,

"dept\_id":0,

"executor":0,

"evaluator":0,

"frequency":"NA",

"searching\_for": "tasksconfiguration",

"loca\_list":[{"loca\_id":95}]

}

Response: JSON String

{

"response": {

"data": {

"search\_list": [

{

"task\_rule\_name": "The Drugs and Cosmetics Rules, 1945",

"task\_impact": "Major",

"task\_impact\_on\_orga": "Moderate",

"task\_frequency\_for\_operation": "Event\_Based",

"task\_activity": "Pay a fee of Rs 150 to obtain a duplicate copy of a license to sell, stock, exhibit or offer for sale or distribute drugs",

"task\_impact\_on\_unit": "Major",

"task\_legi\_name": "The Drugs and Cosmetics Act, 1940",

"task\_activity\_when": "In the case where the original licence is defaced, damaged or lost",

"exec\_id": 294,

"task\_activity\_who": "Person",

"task\_reference": "Rule 59 (3)",

"tmap\_client\_task\_id": "D95100009450",

"task\_lexcare\_id": "R0047445"

}

]

},

"status": "Success"

}

}

Description: search task for configuration page

Function :

TasksConfigurationController.searchConfiguartionPage()

---------------------------------------------------------------------

---------------------------------------------------------------------

* PATH: /getUserDefinedTask Method: POST

Input: [http://localhost:8082/getUserDefinedTask?pagination=true HYPERLINK "http://localhost:8082/getUserDefinedTask?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/getUserDefinedTask?pagination=true HYPERLINK "http://localhost:8082/getUserDefinedTask?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getUserDefinedTask?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/getUserDefinedTask?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getUserDefinedTask?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/getUserDefinedTask?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getUserDefinedTask?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/getUserDefinedTask?pagination=true HYPERLINK "http://localhost:8082/getUserDefinedTask?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getUserDefinedTask?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/getUserDefinedTask?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getUserDefinedTask?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/getUserDefinedTask?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/getUserDefinedTask?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/getUserDefinedTask?pagination=true HYPERLINK "http://localhost:8082/getUserDefinedTask?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getUserDefinedTask?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/getUserDefinedTask?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getUserDefinedTask?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/getUserDefinedTask?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getUserDefinedTask?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/getUserDefinedTask?pagination=true HYPERLINK "http://localhost:8082/getUserDefinedTask?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getUserDefinedTask?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/getUserDefinedTask?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getUserDefinedTask?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/getUserDefinedTask?pagination=true&page=1&per\_page=10"per\_page=10](http://localhost:8082/getUserDefinedTask?pagination=true&page=1&per_page=10)

Response: JSON String

{

     "response": {

         "data": {

             "user\_defined\_tasks": {

                 "Entity": [

                    {

                        "orga\_name": "Emcure Pharmaceuticals Limited",

                        "orga\_id": 2

                    }

                ],

                "Function": [

                    {

                        "loca\_id": 2,

                        "orga\_id": 2,

                        "dept\_name": "HR",

                        "dept\_id": 3

                    }

                ],

                "Executor": [

                    {

                        "loca\_id": 2,

                        "user\_id": 1,

                        "user\_name": "Prashant1 Kolage1",

                        "orga\_id": 2,

                        "dept\_id": 3

                    }

                ],

                "Legislation": [

                    {

                        "task\_legi\_id": 58,

                        "task\_legi\_name": "The Electricity Act, 2003"

                    }

                ],

                "Rule": [

                    {

                        "task\_legi\_id": 58,

                        "task\_rule\_name": "The Central Electricity Authority",

                        "task\_rule\_id": 1059

                    }

                ],

                "Unit": [

                    {

                        "loca\_id": 2,

                        "loca\_name": "Pune",

                        "orga\_id": 2

                    }

                ],

                "Evaluator": [

                    {

                        "loca\_id": 2,

                        "user\_id": 1,

                        "user\_name": "Prashant1 Kolage1",

                        "orga\_id": 2,

                        "dept\_id": 3

                    }

                ],

                "SUBTASK": [

                    {

                        "task\_legi\_id": 58,

                        "task\_rule\_name": "The Central Electricity Authority",

                        "task\_impact": "Major",

                        "eval\_id": 1,

                        "task\_configured\_frequency": "User\_Defined",

                        "task\_activity": "Provide a detailed Safety Manual",

                        "orga\_id": 2,

                        "task\_legi\_name": "The Electricity Act, 2003",

                        "clientTaskId": "D27050000540",

                        "exec\_id": 1,

                        "loca\_id": 2,

                        "task\_reference": "Regulation 5",

                        "task\_who": "Owner",

                        "task\_rule\_id": 1059,

                        "task\_procedure": "NA",

                        "dept\_id": 3,

                        "task\_when": "On applicability of the act",

                        "task\_frequency": "One\_Time"

                    }

                 ]

             }

         },

         "meta": {

             "path": "<http://localhost:8082/getUserDefinedTask>",

             "per\_page": 10,

             "total": 1,

             "last\_page": 1,

             "from": 1,

             "to": 10,

             "current\_page": 1

         },

         "links": {

             "next": "<http://localhost:8082/getUserDefinedTask?page=1>",

             "last": "<http://localhost:8082/getUserDefinedTask?page=1>",

             "prev": "<http://localhost:8082/getUserDefinedTask?page=1>",

             "first": "<http://localhost:8082/getUserDefinedTask?page=1>"

         },

         "status": "Success"

     }

}

Description: get the user defined task

Function : SubTaskController.getUserDefinedTask()

* PATH: /importsubtask Method: POST Response: JSON String

Description: import sub tasks

Function : SubTaskController.importsubtask()

Status: Function Not Available

* PATH: /getimportedsubtask Method: POST

Input : [http://localhost:8082/getimportedsubtask?pagination=true HYPERLINK "http://localhost:8082/getimportedsubtask?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/getimportedsubtask?pagination=true HYPERLINK "http://localhost:8082/getimportedsubtask?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getimportedsubtask?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/getimportedsubtask?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getimportedsubtask?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/getimportedsubtask?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getimportedsubtask?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/getimportedsubtask?pagination=true HYPERLINK "http://localhost:8082/getimportedsubtask?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getimportedsubtask?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/getimportedsubtask?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getimportedsubtask?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/getimportedsubtask?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/getimportedsubtask?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/getimportedsubtask?pagination=true HYPERLINK "http://localhost:8082/getimportedsubtask?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getimportedsubtask?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/getimportedsubtask?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getimportedsubtask?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/getimportedsubtask?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getimportedsubtask?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/getimportedsubtask?pagination=true HYPERLINK "http://localhost:8082/getimportedsubtask?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getimportedsubtask?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/getimportedsubtask?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getimportedsubtask?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/getimportedsubtask?pagination=true&page=1&per\_page=10"per\_page=10](http://localhost:8082/getimportedsubtask?pagination=true&page=1&per_page=10)

Response: JSON String

{  
 "data": {  
 "imported\_tasks": {  
 "Equip\_Type": [  
 {  
 "equip\_type": "No"  
 }  
 ],  
 "Equip\_Number": [  
 {  
 "equip\_type": "No",  
 "equip\_number": "2"  
 }  
 ],  
 "Entity": [  
 {  
 "orga\_name": "Emcure Pharmaceuticals Limited",  
 "orga\_id": 2  
 }  
 ],  
 "Function": [  
 {  
 "loca\_id": 2,  
 "orga\_id": 2,  
 "dept\_name": "HR",  
 "dept\_id": 3  
 }  
 ],  
 "Executor": [  
 {  
 "loca\_id": 2,  
 "user\_id": 1,  
 "user\_name": "Prashant1 Kolage1",  
 "orga\_id": 2,  
 "dept\_id": 3  
 }  
 ],  
 "Equip\_Freq": [  
 {  
 "equip\_loca": "Pune",  
 "equip\_Freq": "NO",  
 "equip\_type": "No",  
 "equip\_number": "2"  
 }  
 ],  
 "Legislation": [  
 {  
 "task\_legi\_id": 135,  
 "task\_legi\_name": "The Companies Act, 2013"  
 }  
 ],  
 "Equip\_Loca": [  
 {  
 "equip\_loca": "Pune",  
 "equip\_type": "No",  
 "equip\_number": "2"  
 }  
 ],  
 "ImportedSubTask": [  
 {  
 "task\_legi\_id": 135,  
 "task\_rule\_name": "The Companies (Audit and Auditors) Rules, 2014",  
 "task\_impact": "Low",  
 "equi\_type": "No",  
 "equi\_freq": "NO",  
 "equi\_last\_examination\_date": "2019-06-16 12:52:52",  
 "eval\_id": 1,  
 "equi\_desc": "Testing sub task description",  
 "task\_activity": "Ensure that the auditor complies with all the applicable accounting standards",  
 "orga\_id": 2,  
 "task\_legi\_name": "The Companies Act, 2013",  
 "equi\_sub\_task\_id": "1",  
 "clientTaskId": "D02030000003",  
 "exec\_id": 1,  
 "loca\_id": 2,  
 "task\_reference": "Section 143(9)",  
 "task\_who": "Company",  
 "equi\_number": "2",  
 "equi\_loca": "Pune",  
 "task\_rule\_id": 830,  
 "task\_procedure": "",  
 "dept\_id": 3,  
 "task\_when": "On the applicability of the Act",  
 "task\_frequency": "Weekly"  
 }  
 ],  
 "Rule": [  
 {  
 "task\_legi\_id": 135,  
 "task\_rule\_name": "The Companies (Audit and Auditors) Rules, 2014",  
 "task\_rule\_id": 830  
 }  
 ],  
 "Unit": [  
 {  
 "loca\_id": 2,  
 "loca\_name": "Balewadi",  
 "orga\_id": 2  
 }  
 ],  
 "Evaluator": [  
 {  
 "loca\_id": 2,  
 "user\_id": 1,  
 "user\_name": "Prashant1 Kolage1",  
 "orga\_id": 2,  
 "dept\_id": 3  
 }  
 ]  
 }  
 },  
 "meta": {  
 "path": "<http://localhost:9002/compliance/getimportedsubtask>",  
 "per\_page": 10,  
 "total": 1,  
 "last\_page": 1,  
 "from": 1,  
 "to": 10,  
 "current\_page": 1  
 },  
 "links": {  
 "next": "<http://localhost:9002/compliance/getimportedsubtask?page=1>",  
 "last": "<http://localhost:9002/compliance/getimportedsubtask?page=1>",  
 "prev": "<http://localhost:9002/compliance/getimportedsubtask?page=1>",  
 "first": "<http://localhost:9002/compliance/getimportedsubtask?page=1>"  
 },  
 "status": "Success"  
}

Description: get imported sub tasks

Function : SubTaskController.getimportedsubtask()

* PATH: /getsubtasksforconfiguration Method: POST

Input: [http://localhost:8082/getsubtasksforconfiguration?pagination=true HYPERLINK "http://localhost:8082/getsubtasksforconfiguration?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/getsubtasksforconfiguration?pagination=true HYPERLINK "http://localhost:8082/getsubtasksforconfiguration?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getsubtasksforconfiguration?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/getsubtasksforconfiguration?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getsubtasksforconfiguration?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/getsubtasksforconfiguration?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getsubtasksforconfiguration?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/getsubtasksforconfiguration?pagination=true HYPERLINK "http://localhost:8082/getsubtasksforconfiguration?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getsubtasksforconfiguration?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/getsubtasksforconfiguration?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getsubtasksforconfiguration?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/getsubtasksforconfiguration?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/getsubtasksforconfiguration?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/getsubtasksforconfiguration?pagination=true HYPERLINK "http://localhost:8082/getsubtasksforconfiguration?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getsubtasksforconfiguration?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/getsubtasksforconfiguration?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getsubtasksforconfiguration?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/getsubtasksforconfiguration?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getsubtasksforconfiguration?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/getsubtasksforconfiguration?pagination=true HYPERLINK "http://localhost:8082/getsubtasksforconfiguration?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getsubtasksforconfiguration?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/getsubtasksforconfiguration?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getsubtasksforconfiguration?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/getsubtasksforconfiguration?pagination=true&page=1&per\_page=10"per\_page=10](http://localhost:8082/getsubtasksforconfiguration?pagination=true&page=1&per_page=10)

Response: JSON String

{

     "response": {

         "data": {

             "tasks\_for\_configurations": {}

         },

         "status": "Success"

     }

}

Description: get sub task for configuration

Function : SubTaskController.getsubtasksforconfiguration()

* PATH: /savesubtasksconfiguration Method: POST

Input: <http://localhost:9002/compliance/savesubtasksconfiguration>

{

    "ttrn\_next\_examination\_date":"10-05-2021",

    "ttrn\_legal\_due\_date":"10-05-2021",

    "ttrn\_fh\_due\_date":"10-05-2021",

    "ttrn\_pr\_due\_date":"10-05-2021",

    "ttrn\_rw\_due\_date":"10-05-2021",

    "ttrn\_uh\_due\_date":"10-05-2021",

    "ttrn\_document":"1",

    "ttrn\_historical":"1",

    "ttrn\_prior\_days\_buffer":"1",

    "ttrn\_alert\_days":"1",

    "ttrn\_first\_alert":"10-05-2021",

    "ttrn\_second\_alert":"10-05-2021",

    "ttrn\_third\_alert":"10-05-2021",

    "ttrn\_no\_of\_back\_days\_allowed":"1",

    "tasks\_list":[{"ttrn\_client\_task\_id":"D27050000540",

    "ttrn\_sub\_task\_id":"0"}]

}

Response: JSON String

{

    "response": {

        "SuccessIds": [

            {

                "successClientTasksId": "0"

            }

        ],

        "status": "Success"

    }

}

Description: save sub task configuration

Function : SubTaskController.savesubtasksconfiguration()

* PATH: /getconfiguredsubtask Method: POST

Input: [http://localhost:8082/getconfiguredsubtask?pagination=true HYPERLINK "http://localhost:8082/getconfiguredsubtask?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/getconfiguredsubtask?pagination=true HYPERLINK "http://localhost:8082/getconfiguredsubtask?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getconfiguredsubtask?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/getconfiguredsubtask?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getconfiguredsubtask?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/getconfiguredsubtask?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getconfiguredsubtask?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/getconfiguredsubtask?pagination=true HYPERLINK "http://localhost:8082/getconfiguredsubtask?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getconfiguredsubtask?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/getconfiguredsubtask?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getconfiguredsubtask?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/getconfiguredsubtask?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/getconfiguredsubtask?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/getconfiguredsubtask?pagination=true HYPERLINK "http://localhost:8082/getconfiguredsubtask?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getconfiguredsubtask?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/getconfiguredsubtask?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getconfiguredsubtask?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/getconfiguredsubtask?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getconfiguredsubtask?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/getconfiguredsubtask?pagination=true HYPERLINK "http://localhost:8082/getconfiguredsubtask?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getconfiguredsubtask?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/getconfiguredsubtask?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/getconfiguredsubtask?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/getconfiguredsubtask?pagination=true&page=1&per\_page=10"per\_page=10](http://localhost:8082/getconfiguredsubtask?pagination=true&page=1&per_page=10)

Response: JSON String

{

     "response": {

         "data": {

             "configured\_tasks": {}

         },

         "status": "Success"

     }

}

Description: get configured sub task

Function : SubTaskController.getconfiguredsubtask()

* PATH: /updateStatus Method: POST Response: JSON String

Description: activate or deactivate the sub task Function : SubTaskController.updateStatus()

---------------------------------------------------------------------

----------------------------------------------

* PATH: /saveShowCauseNotice Method: POST

Input:

{  
 "scau\_orga\_id":1,  
 "scau\_loca\_id":1,  
 "scau\_dept\_id":1,  
 "scau\_ralated\_to":1,  
 "scau\_comments":"No Comments",  
 "scau\_action\_taken":"",  
 "scau\_next\_action\_item":"No",  
 "scau\_responsible\_person":1,  
 "scau\_reporting\_person":1,  
 "scau\_notice\_date":"19-05-2021",  
 "scau\_received\_date":"19-05-2021",  
 "scau\_deadline\_date":"19-05-2021",  
 "scau\_remainder\_date":"19-05-2021"  
}

Response: JSON String

Description: save showcase notice

Function : ShowCauseNoticeController.saveShowCauseNotice()

* PATH: /getAccessWiseOrgaLocaDept Method: POST

Input: <http://10.21.20.14:8081/compliance/getAccessWiseOrgaLocaDept>

Response: JSON String

Description: get access wise showcase notice

Function :

ShowCauseNoticeController.getAccessWiseOrgaLocaDept()

* PATH: /getAllShowCauseNotice Method: POST

Input: <http://localhost:9002/compliance/getAllShowCauseNotice>

Response: JSON String

{

    "data": {

        "DD\_data": [

            {

                "Entity": [

                    {

                        "orga\_name": "Gennova Biopharmaceuticals Limited",

                        "orga\_id": 3

                    },

                    {

                        "orga\_name": "Emcure Pharmaceuticals Limited",

                        "orga\_id": 2

                    }

                ],

                "Function": [

                    {

                        "loca\_id": 24,

                        "orga\_id": 3,

                        "dept\_name": "EHS",

                        "dept\_id": 7

                    },

                    {

                        "loca\_id": 34,

                        "orga\_id": 2,

                        "dept\_name": "Secretarial",

                        "dept\_id": 24

                    }

                ],

                "Unit": [

                    {

                        "loca\_id": 24,

                        "loca\_name": "Hinjewadi VFC",

                        "orga\_id": 3

                    },

                    {

                        "loca\_id": 34,

                        "loca\_name": "Emcure Pharma- Secreterial",

                        "orga\_id": 2

                    }

                ]

            }

        ],

        "notice\_list": [

            {

                "comments": "Non-compliance for",

                "orga\_name": "Gennova Biopharmaceuticals Limited",

                "orga\_id": 3,

                "action\_taken": "No",

                "reminder\_date": "11-09-2019",

                "dept\_name": "EHS",

                "recieved\_date": "09-09-2019",

                "reporting\_user\_name": "Bharat Dighe",

                "responsible\_user\_id": 44,

                "related\_to": "Non compliace of etp function",

                "loca\_id": 24,

                "deadline\_date": "12-09-2019",

                "notice\_date": "09-09-2019",

                "responsible\_user\_name": "Ganesh Pandit",

                "loca\_name": "Hinjewadi VFC",

                "dept\_id": 7,

                "scau\_id": 1,

                "status": "Open",

                "reporting\_user\_id": 91

            },

            {

                "comments": "",

                "orga\_name": "Emcure Pharmaceuticals Limited",

                "orga\_id": 2,

                "action\_taken": "No",

                "reminder\_date": "25-01-2020",

                "dept\_name": "Secretarial",

                "recieved\_date": "07-01-2020",

                "reporting\_user\_name": "Jayant Prakash",

                "responsible\_user\_id": 316,

                "related\_to": "Non Compliance of Companies Act",

                "loca\_id": 34,

                "deadline\_date": "30-01-2020",

                "notice\_date": "07-01-2020",

                "responsible\_user\_name": "Jayant Prakash",

                "loca\_name": "Emcure Pharma- Secreterial",

                "dept\_id": 24,

                "scau\_id": 2,

                "status": "Open",

                "reporting\_user\_id": 316

            }

        ]

    },

    "status": "Success"

}

Description: get list of all showcase notice

Function : ShowCauseNoticeController.getAllShowCauseNotice()

* PATH: /saveActionItem Method: POST

Input: <http://localhost:9002/compliance/saveActionItem>

{

    "tscn\_scau\_id":2,

    "tscn\_comment":"",

    "tscn\_action\_taken":"",

    "tscn\_next\_action\_item":"",

    "tcau\_status":"",

    "tscn\_next\_due\_date":"",

    "tscn\_reminder\_date":""

}

Response: JSON String

Description: save action item

Function : ShowCauseNoticeController.saveActionItem()

* PATH: /getAllActionItem Method: POST

Input: <http://localhost:9002/compliance/getAllActionItem>

{

"scau\_id": 1

}

Response: JSON String

{

"response": {

"data": {

"action\_list": [

{

"tscn\_status": null,

"doc\_list": [],

"tscn\_next\_action\_item": "1",

"tscn\_comment": "No comment",

"tscn\_id": 1,

"tscn\_action\_taken": "No",

"tscn\_next\_due\_date": "10-09-2019",

"tscn\_reminder\_date": "10-09-2019"

}

]

}

}

}

Description: get all the list of action items

Function : ShowCauseNoticeController.getAllActionItem()

* PATH: /getShowCauseNoticeDetails Method: POST

Input: <http://localhost:9002/compliance/getShowCauseNoticeDetails>

{

    "scau\_id":"1"

}

Response: JSON String

{

    "data": {

        "ShowCauseNoticeDetails": {

            "comments": "Non-compliance for",

            "doc\_list": [

                {

                    "doc\_name": "Qualifications.docx",

                    "doc\_id": 1

                }

            ],

            "orga\_name": "Gennova Biopharmaceuticals Limited",

            "orga\_id": 3,

            "action\_taken": "No",

            "reminder\_date": "11-09-2019",

            "dept\_name": "EHS",

            "recieved\_date": "09-09-2019",

            "reporting\_user\_name": "Bharat Dighe",

            "responsible\_user\_id": 44,

            "related\_to": "Non compliace of etp function",

            "next\_action\_item": "",

            "loca\_id": 24,

            "deadline\_date": "12-09-2019",

            "notice\_date": "09-09-2019",

            "responsible\_user\_name": "Ganesh Pandit",

            "loca\_name": "Hinjewadi VFC",

            "dept\_id": 7,

            "scau\_id": 1,

            "status": "Open",

            "reporting\_user\_id": 91

        }

    },

    "status": "Success"

}

Description: get showcase notice details

Function :

ShowCauseNoticeController.getShowCauseNoticeDetails()

* PATH: /downloadShowCauseDocument Method: POST

Input: /compliance/downloadShowCauseDocument?doc\_id=21

Response: JSON String

Description: download showcase notice

Function :

ShowCauseNoticeController.downloadProofOfCompliance()

* PATH: /updateShowCauseNotice Method: POST

Input:

{

"scau\_id": 1

"scau\_orga\_id":1,

"scau\_loca\_id":1,

"scau\_dept\_id":1,

"scau\_ralated\_to":1,

"scau\_comments":"No Comments",

"scau\_action\_taken":"",

"scau\_next\_action\_item":"No",

"scau\_responsible\_person":1,

"scau\_reporting\_person":1,

"scau\_notice\_date":"19-05-2021",

"scau\_received\_date":"19-05-2021",

"scau\_deadline\_date":"19-05-2021",

"scau\_remainder\_date":"19-05-2021"

}

Response: JSON String

{

    "response": {

        "status": "Success"

    }

}

Description: update showcase notice

Function : ShowCauseNoticeController.updateShowCauseNotice()

* PATH: /getupcomingtasks Method: POST Response: JSON String

Description: get the upcoming tasks by sending an email to executor

Function : SchedularController.getUpcomingTasks()

Status: Function Not Available

* PATH: /getescalationtasksforevaluator Method: POST

Response: JSON String

Description: send escalation email to an evaluator

Function : SchedularController.getEscalationTasksForEvaluator()

Status: Function Not Available

* PATH: /getescalationtasksforfunctionhead Method: POST

Response: JSON String

Description: get escalation task for function head and send an escalation email

Function : SchedularController.getEscalationTasksForFunctionHead()

Status: Function Not Available

* PATH: /getExportDataById Method: POST

Input: <http://localhost:9002/compliance/getExportDataById>

{

"orga\_id": 0,

"loca\_id":0,

"dept\_id":0,

"executor\_id":0,

"evaluator\_id":0,

"fh\_user\_id":0,

"cat\_law\_id":0,

"impact":"null",

"proh\_pres":"null",

"type\_of\_task":"null",

"frequency":"null",

"task\_status":"null",

"event":"null",

"sub\_event":"null",

"legi\_id":0,

"rule\_id":0,

"task\_id":"null"

}

Response: JSON String

{

"response": {

"status": "Success"

}

}

Description: get the data for export operation by id Function : RepositoryController.getExportDataById()

* PATH: /getExportData Method: POST Response: JSON String

Description: export data in file

Function : RepositoryController.getExportData()

Status: Function Not Available

* PATH: /getoverallCountForDashboard Method: POST

Response: JSON String

Description: get total count for dashboard page Function : NewDashboardController.getOverallCount()

Status: Function Not Available

* PATH: /getoveralldrilleddata Method: GET

Response: JSON String

Description: get the drilled down data

Function : NewDashboardController.getoveralldrilleddata()

Status: Function Not Available

* PATH: /getEntityRisksCount Method: POST

Response: JSON String

Description: get entity risk count

Function : NewDashboardController.getEntityRisksCount()

Status: Function Not Available

* PATH: /getEntityRisksModalData Method: GET

Response: JSON String

Description: get entity risk drilled down data

Function : NewDashboardController.getEntityRisksDataOnModal()

Status: Function Not Available

* PATH: /getUnitRisksCount Method: POST

Response: JSON String Description: get unit risks count

Function : NewDashboardController.getUnitRisksCount()

Status: Function Not Available

* PATH: /getUnitRisksModalData Method: GET

Response: JSON String

Description: get unit risks model data

Function : NewDashboardController.getUnitRisksModalData()

Status: Function Not Available

* PATH: /getFunctionRisksCount Method: POST

Response: JSON String

Description: get unit risks model data

Function : NewDashboardController.getFunctionRisksCount()

Status: Function Not Available

* PATH: /getFunctionRisksModalData Method: GET

Response: JSON String

Description: get data for function risks on selection Function : NewDashboardController.getFunctionRisksModalData()

Status: Function Not Available

* PATH: /getLocationListByOrgaId Method: POST

Response: JSON String

Description: get location data by organization id

Function : NewDashboardController.getLocationListByOrgaId()

Status: Function Not Available

* PATH: /getFinancialRisksCount Method: POST

Response: JSON String

Description: get financial risk count

Function : NewDashboardController.getFinancialRisksCount()

Status: Function Not Available

* PATH: /filterDataByOrgaIdAndUnitIdURL Method: POST

Response: JSON String

Description: filter the data by org id and unit id

Function :

NewDashboardController.filterDataByOrgaIdAndUnitIdURL()

Status: Function Not Available

---------------------------------------------------------------------

----------------------------------------------

* PATH: /importlegalupdates Method: POST

Response: JSON String

Description: import the legal updates

Function : LegalUpdateController.importLegalUpdates()

Status: Function Not Available

---------------------------------------------------------------------

---------------------------------------------------------------------

* PATH:/functions/all Method: GET

Input: [http://localhost:8082//functions/all?pagination=true HYPERLINK "http://localhost:8082//functions/all?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082//functions/all?pagination=true HYPERLINK "http://localhost:8082//functions/all?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082//functions/all?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082//functions/all?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082//functions/all?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082//functions/all?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082//functions/all?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082//functions/all?pagination=true HYPERLINK "http://localhost:8082//functions/all?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082//functions/all?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082//functions/all?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082//functions/all?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082//functions/all?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082//functions/all?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082//functions/all?pagination=true HYPERLINK "http://localhost:8082//functions/all?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082//functions/all?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082//functions/all?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082//functions/all?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082//functions/all?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082//functions/all?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082//functions/all?pagination=true HYPERLINK "http://localhost:8082//functions/all?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082//functions/all?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082//functions/all?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082//functions/all?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082//functions/all?pagination=true&page=1&per\_page=10"per\_page=10](http://localhost:8082//functions/all?pagination=true&page=1&per_page=10)

Response: JSON String

{

     "response": {

         "data": [

             {

                 "dept\_name": "HR",

                 "dept\_id": "3"

             }

         ],

         "meta": {

             "path": "<http://localhost:8082/functions/all>",

             "per\_page": 10,

             "total": 1,

             "last\_page": 1,

             "from": 1,

             "to": 10,

             "current\_page": 1

         },

         "links": {

             "next": "<http://localhost:8082/functions/all?page=1>",

             "last": "<http://localhost:8082/functions/all?page=1>",

             "prev": "<http://localhost:8082/functions/all?page=1>",

             "first": "<http://localhost:8082/functions/all?page=1>"

         }

     }

}

Description: get the list of functions Function : FunctionController.listFunctions()

* PATH: /savefunction Method: POST

Input: <http://127.0.0.1:8082/savefunction?dept_name=HR1>

Response: JSON String

{

     "response": {

         "data": {

             "department\_data": {

                 "dept\_name": "HR1"

             }

         },

         "message": "Function Saved",

         "status": "Success"

     }

}

Description: save the function in db Function : FunctionController.saveFunction()

* PATH: /editfunction Method: POST Response: JSON String

{

     "response": {

         "data": {

             "dept\_data": {

                 "dept\_id": 1,

                 "dept\_name": "Finance",

                 "dept\_parent\_id": 0,

                 "dept\_approval\_status": "1",

                "dept\_enable\_status": "1",

                 "dept\_added\_by": 1,

                 "dept\_created\_at": "17-06-2019 10:41:29",

                 "is\_deleted": **false**

             }

         },

         "status": "Success"

     }

}

Description: edit function

Function : FunctionController.editFunction()

* PATH: /updatefunction Method: POST

Input: <http://127.0.0.1:8082/updatefunction/4?dept_name=HR2>

Response: JSON String

{

     "response": {

         "message": "Function Updated",

         "status": "Success"

     }

}

Description: update functions

Function : FunctionController.updateFunction()

* PATH: /deletefunction Method: POST

Input: <http://127.0.0.1:8082/deleteFunction/4>

Response: JSON String

{

     "response": {

         "message": "Function Deleted",

         "status": "Success"

     }

}

Description: delete the function

Function : FunctionController.deleteFunction()

* PATH: /isDeptNameExist Method: POST Response: JSON String

{

     "response": {

         "data": {

             "function\_data": {

                 "is\_exists\_by\_name": **false**

             }

         },

         "status": "Success"

     }

}

Description: checks if department name exists Function : FunctionController.isDeptNameExist()

* PATH: /importfunction Method: POST

Input : /compliance/importfunction

Response: JSON String

{

"response": {

"status": "Success"

}

}

Description: import function

Function : FunctionController.importLegalUpdates()

---------------------------------------------------------------------

---------------------------------------------------------------------

* PATH: /listentitiesmapping Method: GET

Input: [http://localhost:8082/listentitiesmapping?pagination=true HYPERLINK "http://localhost:8082/listentitiesmapping?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/listentitiesmapping?pagination=true HYPERLINK "http://localhost:8082/listentitiesmapping?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listentitiesmapping?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/listentitiesmapping?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listentitiesmapping?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/listentitiesmapping?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listentitiesmapping?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/listentitiesmapping?pagination=true HYPERLINK "http://localhost:8082/listentitiesmapping?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listentitiesmapping?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/listentitiesmapping?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listentitiesmapping?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/listentitiesmapping?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/listentitiesmapping?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/listentitiesmapping?pagination=true HYPERLINK "http://localhost:8082/listentitiesmapping?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listentitiesmapping?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/listentitiesmapping?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listentitiesmapping?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/listentitiesmapping?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listentitiesmapping?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/listentitiesmapping?pagination=true HYPERLINK "http://localhost:8082/listentitiesmapping?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listentitiesmapping?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/listentitiesmapping?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listentitiesmapping?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/listentitiesmapping?pagination=true&page=1&per\_page=10"per\_page=10](http://localhost:8082/listentitiesmapping?pagination=true&page=1&per_page=10)

Response: JSON String

{

     "data": [

         {

             "enti\_loca\_name": "Hinjewadi EOU",

             "enti\_id": "1",

             "enti\_dept\_name": "HR",

             "enti\_approval\_status": "1",

             "enti\_enable\_status": "1",

             "enti\_orga\_name": "Emcure Pharmaceuticals Limited"

         }

     ],

     "meta": {

         "path": "<http://localhost:8082/listentitiesmapping>",

         "per\_page": 10,

         "total": 1,

         "last\_page": 1,

         "from": 1,

         "to": 10,

         "current\_page": 1

     },

     "links": {

         "next": "<http://localhost:8082/listentitiesmapping?page=1>",

         "last": "<http://localhost:8082/listentitiesmapping?page=1>",

         "prev": "<http://localhost:8082/listentitiesmapping?page=1>",

         "first": "<http://localhost:8082/listentitiesmapping?page=1>"

     }

}

Description: listing the entity mapping

Function : EntityMappingController.listEntitiesMapping()

* PATH: /saveentitiesmapping Method: POST

Response: JSON String Description: save entity mapping

Function : EntityMappingController.saveEntitiesMapping()

* PATH: /getMappedDepartments Method: POST

Input : /getMappedDepartments?enti\_orga\_id=3&enti\_loca\_id=1

Response: JSON String

{

     "response": {

        "data": {

            "data\_list": {

                 "unmappedDepartments": [

                    {

                        "dept\_name": "Finance",

                        "dept\_id": "1"

                    },

                    {

                        "dept\_name": "HR",

                        "dept\_id": "3"

                    },

                    {

                        "dept\_name": "HR2",

                        "dept\_id": "4"

                    },

                    {

                        "dept\_name": "Stores",

                        "dept\_id": "2"

                    }

                ],

                "mappedDepartments": []

             }

         },

        "status": "Success"

     }

}

Description: get all mapped departments

Function : EntityMappingController.getAllDepartments()

* PATH: /getAllMapping Method: POST

Input: /getMappedDepartments?enti\_orga\_id=1&enti\_loca\_id=1

Response: JSON String

{

     "response": {

         "data": {

             "data\_list": {

                 "unmappedDepartments": [

                    {

                        "dept\_name": "Finance",

                        "dept\_id": "1"

                    },

                    {

                        "dept\_name": "HR",

                        "dept\_id": "3"

                    }

                 ],

                 "mappedDepartments": [

                    {

                        "dept\_name": "Stores",

                        "dept\_id": "2"

                    }

                  ]

             }

         },

         "status": "Success"

     }

}

Description: get all mapping in list

Function : EntityMappingController.getAllMapping()

* PATH: /getAllMappingReport Method: POST

Input:

<http://localhost:9002/compliance/getAllMappingReport>

Response: JSON String

{

"response": {

"data": {

"data\_list": {

"Entity": [

{

"orga\_name": "Gennova Biopharmaceuticals Limited",

"orga\_id": 3

}

],

"Function": [

{

"loca\_id": 1,

"orga\_id": 3,

"dept\_name": "HR6",

"dept\_id": 8

}

],

"Unit": [

{

"loca\_id": 1,

"loca\_name": "Hinjewadi EOU",

"orga\_id": 3

}

]

}

},

"status": "Success"

}

}

Description: get all mapping reports

Function : EntityMappingController.getAllMappingReport()

* PATH: /importentitymappingfromfile Method: POST

Response: JSON String

Description: import entity mapping by uploading the data Function : EntityMappingController.importEntityMapping()

Status: Input File Format Required

---------------------------------------------------------------------

----------------------------------------------

* PATH: /listentities Method: GET

Input: [http://localhost:8082/listentities?pagination=true HYPERLINK "http://localhost:8082/listentities?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/listentities?pagination=true HYPERLINK "http://localhost:8082/listentities?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listentities?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/listentities?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listentities?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/listentities?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listentities?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/listentities?pagination=true HYPERLINK "http://localhost:8082/listentities?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listentities?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/listentities?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listentities?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/listentities?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/listentities?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/listentities?pagination=true HYPERLINK "http://localhost:8082/listentities?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listentities?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/listentities?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listentities?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/listentities?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listentities?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/listentities?pagination=true HYPERLINK "http://localhost:8082/listentities?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listentities?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/listentities?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listentities?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/listentities?pagination=true&page=1&per\_page=10"per\_page=10](http://localhost:8082/listentities?pagination=true&page=1&per_page=10)

Response: JSON String

{

     "data": [

         {

             "orga\_parent\_id": 1,

             "orga\_name": "Emcure Pharmaceuticals Limited",

             "orga\_id": 2,

             "orga\_parent\_name": "NA"

         }

     ],

     "meta": {

         "path": "<http://localhost:8082/listentities>",

         "per\_page": 10,

         "total": 1,

         "last\_page": 1,

         "from": 1,

         "to": 10,

         "current\_page": 1

     },

     "links": {

         "next": "<http://localhost:8082/listentities?page=1>",

         "last": "<http://localhost:8082/listentities?page=1>",

         "prev": "<http://localhost:8082/listentities?page=1>",

         "first": "<http://localhost:8082/listentities?page=1>"

     }

}

Description: get single entity

Function : EntityController.listEntities()

* PATH: /listallentities Method: GET

Input: <http://127.0.0.1:8082/listallentities?pagination=true>

Response: JSON String

{

     "response": {

         "data": [

             {

                 "orga\_parent\_id": 1,

                 "orga\_name": "Emcure Pharmaceuticals Limited",

                 "orga\_id": 2,

                 "orga\_parent\_name": "NA"

             },

             {

                 "orga\_parent\_id": 1,

                 "orga\_name": "Gennova Biopharmaceuticals Limited",

                 "orga\_id": 3,

                 "orga\_parent\_name": "NA"

             }

         ],

         "meta": {

             "path": "<http://127.0.0.1:8082/listallentities>",

             "per\_page": 5,

             "total": 2,

             "last\_page": 1,

             "from": 1,

             "to": 5,

             "current\_page": 1

         },

         "links": {

             "next": "<http://127.0.0.1:8082/listallentities?page=1>",

             "last": "<http://127.0.0.1:8082/listallentities?page=1>",

             "prev": "<http://127.0.0.1:8082/listallentities?page=1>",

             "first": "<http://127.0.0.1:8082/listallentities?page=1>"

          }

      }

}

Description: get the list of all entities Function : EntityController.listAllEntities()

* PATH: /listAllForAddingEntity Method: GET

Input: [http://localhost:8082/listAllForAddingEntity?pagination=true HYPERLINK "http://localhost:8082/listAllForAddingEntity?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/listAllForAddingEntity?pagination=true HYPERLINK "http://localhost:8082/listAllForAddingEntity?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listAllForAddingEntity?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/listAllForAddingEntity?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listAllForAddingEntity?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/listAllForAddingEntity?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listAllForAddingEntity?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/listAllForAddingEntity?pagination=true HYPERLINK "http://localhost:8082/listAllForAddingEntity?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listAllForAddingEntity?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/listAllForAddingEntity?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listAllForAddingEntity?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/listAllForAddingEntity?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/listAllForAddingEntity?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/listAllForAddingEntity?pagination=true HYPERLINK "http://localhost:8082/listAllForAddingEntity?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listAllForAddingEntity?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/listAllForAddingEntity?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listAllForAddingEntity?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/listAllForAddingEntity?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listAllForAddingEntity?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/listAllForAddingEntity?pagination=true HYPERLINK "http://localhost:8082/listAllForAddingEntity?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listAllForAddingEntity?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/listAllForAddingEntity?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listAllForAddingEntity?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/listAllForAddingEntity?pagination=true&page=1&per\_page=10"per\_page=10](http://localhost:8082/listAllForAddingEntity?pagination=true&page=1&per_page=10)

Response: JSON String

{

     "response": {

        "data": {

            "entity\_list": [

                {

                    "orga\_id": 7,

                    "orga\_name": "Emcure Pharmaceuticals Co. Ltd. Shanghai (China)",

                    "orga\_parent\_id": 1,

                    "orga\_approval\_status": "1",

                    "orga\_enable\_status": "1",

                    "orga\_added\_by": 1,

                    "orga\_created\_at": "08-04-2021 11:16:30",

                    "is\_deleted": **false**

                },

                {

                    "orga\_id": 6,

                    "orga\_name": "Emcure Pharmaceuticals Pty. Ltd. Australia",

                    "orga\_parent\_id": 1,

                    "orga\_approval\_status": "1",

                    "orga\_enable\_status": "1",

                    "orga\_added\_by": 1,

                    "orga\_created\_at": "08-04-2021 11:16:30",

                    "is\_deleted": **false**

                }

             ]

         },

          "meta": {

             "path": "<http://localhost:8082/listAllForAddingEntity>",

             "per\_page": 10,

             "total": 2,

             "last\_page": 1,

             "from": 1,

             "to": 10,

             "current\_page": 1

         },

         "links": {

             "next": "<http://localhost:8082/listAllForAddingEntity?page=1>",

             "last": "<http://localhost:8082/listAllForAddingEntity?page=1>",

             "prev": "<http://localhost:8082/listAllForAddingEntity?page=1>",

             "first": "<http://localhost:8082/listAllForAddingEntity?page=1>"

         },

         "status": "Success"

     }

}

Description: get the list for adding the entity Function : EntityController.listAllForAddingEntity()

* PATH: /saveentity Method: POST

Input:/saveentity?orga\_parent\_id=1&orga\_name=Emcure

Response: JSON String

{

     "response": {

         "data": {

             "orga\_name": "Emcure"

         },

         "message": "Entity is saved",

         "status": "Success"

     }}

Description: saves the organization/entity Function : x.saveEntity()

* PATH: /updateentity Method: POST

Input: <http://127.0.0.1:8082/updateentity/4>

{

    "orga\_name":"Emcure",

    "orga\_added\_by":1,

    "orga\_parent\_id":1

}

Response: JSON String

{

     "response": {

         "message": "Entity updated",

         "status": "Success"

     }

}

Description: updates the organization/entity Function : EntityController.updateEntity()

* PATH: /isOrgaNameExist Method: POST Response: JSON String

{

     "response": {

         "data": {

             "is\_entity\_exists": **true**

         },

         "status": "Success"

     }

}

Description: checks if the organization/entity exists Function : EntityController.isOrgaNameExist()

* PATH: /importentity Method: POST

Input: /isOrgaNameExist?org\_id=4&org\_name=Emcure

Response: JSON String

{

     "response": {

         "data": {

             "is\_entity\_exists": **true**

         },

         "status": "Success"

     }

}

Description: import entity

Function : EntityController.importLegalUpdates()

---------------------------------------------------------------------

----------------------------------------------

* PATH: /listdesignations Method: GET

Input: [http://localhost:8082/listdesignations?pagination=true HYPERLINK "http://localhost:8082/listdesignations?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/listdesignations?pagination=true HYPERLINK "http://localhost:8082/listdesignations?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listdesignations?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/listdesignations?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listdesignations?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/listdesignations?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listdesignations?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/listdesignations?pagination=true HYPERLINK "http://localhost:8082/listdesignations?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listdesignations?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/listdesignations?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listdesignations?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/listdesignations?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/listdesignations?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/listdesignations?pagination=true HYPERLINK "http://localhost:8082/listdesignations?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listdesignations?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/listdesignations?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listdesignations?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/listdesignations?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listdesignations?pagination=true&page=1&per\_page=10" HYPERLINK "http://localhost:8082/listdesignations?pagination=true HYPERLINK "http://localhost:8082/listdesignations?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listdesignations?pagination=true&page=1&per\_page=10"page=1 HYPERLINK "http://localhost:8082/listdesignations?pagination=true&page=1&per\_page=10"& HYPERLINK "http://localhost:8082/listdesignations?pagination=true&page=1&per\_page=10"per\_page=10" HYPERLINK "http://localhost:8082/listdesignations?pagination=true&page=1&per\_page=10"per\_page=10](http://localhost:8082/listdesignations?pagination=true&page=1&per_page=10)

Response: JSON String

{

     "data": [

         {

             "desi\_id": 1,

             "desi\_name": "Manager",

             "desi\_parent\_id": 0,

             "desi\_approval\_status": "1",

             "desi\_enable\_status": "1",

             "desi\_added\_by": 1,

             "desi\_created\_at": "17-06-2019 10:49:44",

             "is\_deleted": **false**

         },

         {

             "desi\_id": 2,

             "desi\_name": "Developer",

             "desi\_parent\_id": 0,

             "desi\_approval\_status": "1",

              "desi\_enable\_status": "1",

             "desi\_added\_by": 1,

             "desi\_created\_at": "08-04-2021 11:35:16",

              "is\_deleted": **false**

         },

         {

              "desi\_id": 3,

             "desi\_name": "HR",

             "desi\_parent\_id": 9,

             "desi\_approval\_status": "1",

             "desi\_enable\_status": "1",

             "desi\_added\_by": 1,

             "desi\_created\_at": "08-04-2021 11:35:16",

             "is\_deleted": **true**

         }

     ],

     "meta": {

         "path": "<http://localhost:8082/listdesignations>",

         "per\_page": 10,

         "total": 3,

         "last\_page": 1,

         "from": 1,

         "to": 10,

         "current\_page": 1

     },

     "links": {

         "next": "<http://localhost:8082/listdesignations?page=1>",

         "last": "<http://localhost:8082/listdesignations?page=1>",

         "prev": "<http://localhost:8082/listdesignations?page=1>",

         "first": "<http://localhost:8082/listdesignations?page=1>"

     }

}

Description: get the list of designationsFunction : DesignationController.listDesignations()

* PATH: /savedesignations Method: POST

Input :/savedesignations?designation\_name=Developer

Response: JSON String

{

     "response": {

         "data": {

             "designation\_name": "Developer"

         },

         "message": "Designation saved successfully",

         "status": "Success"

     }

}

Description: saves designations

Function : DesignationController.savedesignations()

* PATH: /editdesignations Method: POST

Input: <http://127.0.0.1:8082/editdesignations/2>

Response: JSON String

{

     "response": {

         "data": {

             "designation\_data": {

                 "desi\_id": 2,

                 "desi\_name": "Developer",

                 "desi\_parent\_id": 0,

                 "desi\_approval\_status": "1",

                 "desi\_enable\_status": "1",

                 "desi\_added\_by": 1,

                 "desi\_created\_at": "08-04-2021 11:35:16",

                 "is\_deleted": **null**

             }

         },

         "status": "Success"

     }

}

Description: edit designations

Function : DesignationController.editdesignations()

* PATH: /updatedesignation Method: POST

Input: <http://127.0.0.1:8082/updatedesignation/2>

{

     "designation\_name":"Developer"

}

Response: JSON String

{

     "response": {

         "data": {

             "designation\_data": {

                 "desi\_name": "Developer",

                 "desi\_id": 2

             }

         },

         "message": "Designation Updated",

         "status": "Success"

     }

}

Description: updates the designations

Function : DesignationController.updateFunction()

* PATH: /deletedesignation Method: POST

Input: <http://127.0.0.1:8082/deleteDesignation/3>

Response: JSON String

{

     "response": {

         "message": "Designation Deleted",

         "status": "Success"

     }

}

Description: deletes the designations

Function : DesignationController.deletedesignation()

* PATH: /isDesiNameExist Method: POST

Input: /isDesiNameExist/2?desi\_name=Developer

Response: JSON String

{

     "response": {

         "data": {

             "designation\_data": {

                 "is\_designation\_exists": **true**

             }

         },

         "status": "Success"

      }

}

Description: checks if the designations name exists or not Function : DesignationController.isDesiNameExist()

---------------------------------------------------------------------

----------------------------------------------

* PATH: /getoverallcompliancestatus Method: POST

Input: /compliance/getoverallcompliancestatus

Response: JSON String

{

"data": {

"PosingRisk": 63,

"ReOpened": 0,

"Delayed": 0,

"totalactivetasks": 4010,

"delayed\_reported": 0,

"taskList": [

{

"date": "2021-03-29",

"ttrn\_legal\_due\_date": "29-03-2021",

"ttrn\_rw\_due\_date": "23-03-2021",

"orga\_name": "Emcure Pharmaceuticals Limited",

"task\_executor": "Sheetal Kulkarni",

"task\_activity": "Give a true and fair view of the state of affairs of the Company or companies by complying with the accounting standards",

"task\_legi\_name": "The Companies Act, 2013",

"loca\_id": 34,

"task\_cat\_law": "Corporate",

"task\_who": "Company",

"loca\_name": "Emcure Pharma- Secreterial",

"task\_rule\_name": "The Companies (Accounts) Rules, 2014",

"comments": "",

"ttrn\_frequency\_for\_operation": "Yearly",

"orga\_id": 2,

"ttrn\_pr\_due\_date": "21-03-2021",

"dept\_name": "Secretarial",

"tsk\_impact": "Major",

"ttrn\_client\_task\_id": "D34240000718",

"task\_reference": "Section 129(1)",

"ttrn\_uh\_due\_date": "27-03-2021",

"dept\_id": 24,

"task\_type": "Main",

"task\_when": "In the event of preparing the financial statements",

"ttrn\_fh\_due\_date": "25-03-2021",

"status": "noncomplied"

},

{

"date": "2021-05-30",

"ttrn\_legal\_due\_date": "30-05-2021",

"ttrn\_rw\_due\_date": "26-05-2021",

"orga\_name": "Zuventus Healthcare Limited",

"task\_executor": "Somnath Ghundare",

"task\_activity": "Furnish an annual information return in Form 61A on or before 31st May immediately following the financial year in which the transaction is registered or recorded",

"task\_legi\_name": "The Income Tax Act, 1961 ",

"loca\_id": 23,

"task\_cat\_law": "Taxation- Direct & Indirect",

"task\_who": "Assessee",

"loca\_name": "Zuventus Bhosari",

"task\_rule\_name": " The Income Tax Rules, 1962 ",

"comments": "",

"ttrn\_frequency\_for\_operation": "Yearly",

"orga\_id": 4,

"ttrn\_pr\_due\_date": "22-05-2021",

"dept\_name": "Finance",

"tsk\_impact": "Low",

"ttrn\_client\_task\_id": "D23010014800",

"task\_reference": "Section 285BA and Rule 114E",

"ttrn\_uh\_due\_date": "30-05-2021",

"dept\_id": 1,

"task\_type": "Main",

"task\_when": "On maintaining books of account or other document of financial transaction",

"ttrn\_fh\_due\_date": "30-05-2021",

"status": "posingrisk"

}

],

"totaltasksinloop": 4010,

"Complied": 0,

"NonComplied": 3945,

"WaitingForApproval": 0,

"totalTasks": 0,

"Pending": 2,

"totalcompletedtasks": 0

},

"status": "Success"

}

Description: get all the complaince details

Function : DashboardController.getOverallComplianceStatus()

* PATH: /getentitywiseComplianceStatus Method: POST

Input: /compliance/getentitywiseComplianceStatus

{

}

Response: JSON String

{

"response": {

"data": {

"Entity": [

"Zuventus Healthcare Limited",

"Gennova Biopharmaceuticals Limited",

"Emcure Pharmaceuticals Pty. Ltd. Australia",

"Emcure Pharmaceuticals Ltd Myanmar",

"Emcure Pharmaceuticals Co. Ltd. Shanghai (China)",

"Emcure Pharmaceuticals Pty. Ltd. Australia",

"NA"

],

"PosingRisk": [

27,

3,

33,

0,

0,

0,

0

],

"Emcure Pharmaceuticals Co. Ltd. Shanghai (China)": [

{

"PosingRisk": 0,

"ReOpened": 0,

"Delayed": 0,

"Complied": 0,

"NonComplied": 0,

"WaitingForApproval": 0,

"Pending": 2

}

],

"ReOpened": [

0,

0,

0,

0,

0,

0,

0

],

"Delayed": [

0,

0,

0,

0,

0,

0,

0

],

"orga\_count": 7,

"Gennova Biopharmaceuticals Limited": [

{

"PosingRisk": 3,

"ReOpened": 0,

"Delayed": 0,

"Complied": 0,

"NonComplied": 506,

"WaitingForApproval": 0,

"Pending": 1

}

],

"Emcure Pharmaceuticals Ltd Myanmar": [

{

"PosingRisk": 0,

"ReOpened": 0,

"Delayed": 0,

"Complied": 0,

"NonComplied": 0,

"WaitingForApproval": 0,

"Pending": 2

}

],

"Emcure Pharmaceuticals Pty. Ltd. Australia": [

{

"PosingRisk": 0,

"ReOpened": 0,

"Delayed": 0,

"Complied": 0,

"NonComplied": 0,

"WaitingForApproval": 0,

"Pending": 2

}

],

"NA": [

{

"PosingRisk": 0,

"ReOpened": 0,

"Delayed": 0,

"Complied": 0,

"NonComplied": 0,

"WaitingForApproval": 0,

"Pending": 2

}

],

"Complied": [

0,

0,

0,

0,

0,

0,

0

],

"Zuventus Healthcare Limited": [

{

"PosingRisk": 27,

"ReOpened": 0,

"Delayed": 0,

"Complied": 0,

"NonComplied": 1288,

"WaitingForApproval": 0,

"Pending": 1

}

],

"NonComplied": [

1288,

506,

2151,

0,

0,

0,

0

],

"WaitingForApproval": [

0,

0,

0,

0,

0,

0,

0

]

},

"status": "Success"

}

}

Description: get the entity wise complaince status

Function : DashboardController.getEntityWiseComplianceStatus()

* PATH: /getunitwiseComplianceStatus Method: POST

Input: /compliance/getunitwiseComplianceStatus

Response: JSON String

Description: get the unit wise complaince status

Function : DashboardController.getUnitWiseComplianceStatus()

* PATH: /getfunctionwiseComplianceStatus Method: POST

Input : /compliance/getfunctionwiseComplianceStatus

{}

Response: JSON String

Description: get the function wise complaince status

Function : DashboardController.getFunctionWiseComplianceStatus()

* PATH: /getgraphdrilldown Method: POST

Response: JSON String

Description: get drilled down reports for graph Function : DashboardController.getGraphDrillDown()

Status: Function Not Available

* PATH: /searchgetoverallcompliancestatus Method: POST

Input: /compliance/ searchgetoverallcompliancestatus

{

}

Response: JSON String

Description: search complaince status for all

Function : DashboardController.searchOverallComplianceStatus()

* PATH: /searchGraph Method: POST

Input: <http://localhost:9002/compliance/searchGraph>

{

"date\_from":"01-01-2020",

"date\_to": "02-01-2020"

}

Response: JSON String

Description: search the graph and returns the result Function : DashboardController.searchGraph()

* PATH: /getExportDrillReport Method: POST

Input: <http://localhost:9002/compliance/getExportDrillReport>

{

"status" : "Complied",

"entity\_name":"0",

"fromDate":"01-01-2020",

"toDate":"01-03-2021"

}

Response: JSON String

{

"response": {

"status": "Success"

}

}

Description: get export drilled report

Function : DashboardController.getExportDrillReport()

* PATH: /downloadExportDrillReport Method: GET

Input: /compliance/downloadExportDrillReport

Response: JSON String

Description: downloads the export drilled down report Function : DashboardController.downloadExportDrillReport()

---------------------------------------------------------------------

* PATH: /generateCertificate Method: POST

Input:

{

"date\_from":"01-01-2020",

"date\_to":"05-01-2020"

}

Response: JSON String

{

"response": {

"data": {

"Task": [

{

"task\_rule\_name": "Master Direction - Export of Goods and Services (Updated as on January 12, 2018)",

"task\_impact": "Major",

"task\_functionhead": "Santosh Nair",

"client\_task\_id": "D33010014577",

"task\_status": "Non-Complied",

"task\_executor": "Madan More",

"task\_activity": "Submit a combined SOFTEX form for all the invoices raised on a particular overseas client, including advance remittances received in a month ",

"task\_legi\_name": "The Foreign Exchange Management Act, 1999 ",

"task\_reasone\_of\_non\_compliance": "",

"task\_submitted\_date": "",

"task\_evaluator": "Santosh Nair",

"task\_comment": "",

"task\_who": "Software Exporter",

"task\_when": "In case of long duration contracts involving series of transmissions",

"task\_legal\_due\_date": "03-01-2020",

"task\_frequency": "Monthly"

}

],

"LegTask": [

{

"task\_rule\_name": "The Companies (Acceptance of Deposits) Rules, 2014",

"task\_legi\_name": "The Companies Act, 2013",

"task\_cat\_law\_name": "Corporate"

} {

"task\_rule\_name": "The Self Certification - cum - Consolidated Annual Return Scheme",

"task\_legi\_name": "The Labour Laws (Gujarat Amendment) Act, 2015",

"task\_cat\_law\_name": "Labour Laws"

}

],

"Notices": []

},

"status": "Success"

}

}

Description: generate certificate

Function : CertificateController.genereateCertificate()

* PATH: /downloadCertificateDetails Method: GET

Input: <http://localhost:9002/compliance/downloadCertificateDetails>

Response: JSON String

{

"response": {

"data": [

{

"certificatePath": "c:/path",

"fname": "Ram Kolage",

"created\_at": "13-05-2021",

"from": "13-05-2021",

"to": "13-05-2021",

"orig\_file\_name": "No Name"

}

],

"status": "Success"

}

}

Description: downloads the certificate details Function : CertificateController.downloadCertificate()

* PATH: /download Method: POST

Input : <http://localhost:9002/compliance/download>

{

"certificateName" :"sagar krios\_24052021\_1621857240454.pdf"

}

Response: JSON String

Description: downloads the certificate

Function : CertificateController.downloadCertificate()

* PATH: /downloadDocuments Method: GET

Input: <http://localhost:9002/compliance/downloadDocuments?certificate=sagar> krios\_24052021\_1621857240454.pdf

Response: JSON String

Description: downloads the complaince proof

Function : CertificateController.downloadProofOfCompliance()

---------------------------------------------------------------------

---------------------------------------------------------------------

* PATH: /showComplianceCalendar Method: POST

Input: <http://localhost:9002/compliance/showComplianceCalendar>

Response: JSON String

{

"response": {

"data": {

"task\_assigned\_to\_user": [

{

"task\_which\_entity\_wise": "Performer Wise",

"task\_orga\_id": 4,

"client\_task\_id": "D35240001423",

"task\_function\_head\_id": 316,

"task\_status": "Non Complied",

"task\_loca\_id": 35,

"task\_activity": "Obtain credit rating for deposits accepted by company in the manner prescribed once in a year and copy of the rating shall sent to the Registrar along with return of deposit in form DPT3",

"calender\_date": "2020-05-04 18:30:00.0",

"task\_dept\_id": 24,

"type": "MainTask",

"performer\_due\_date": "2020-05-04 18:30:00.0",

"task\_reviewer\_id": 316,

"task\_performer\_id": 169,

"task\_frequency": "Yearly"

},

{

"task\_which\_entity\_wise": "Performer Wise",

"task\_orga\_id": 4,

"client\_task\_id": "D35240001423",

"task\_function\_head\_id": 316,

"task\_status": "Non Complied",

"task\_loca\_id": 35,

"task\_activity": "Obtain credit rating for deposits accepted by company in the manner prescribed once in a year and copy of the rating shall sent to the Registrar along with return of deposit in form DPT3",

"calender\_date": "2021-05-04 18:30:00.0",

"task\_dept\_id": 24,

"type": "MainTask",

"performer\_due\_date": "2021-05-04 18:30:00.0",

"task\_reviewer\_id": 316,

"task\_performer\_id": 169,

"task\_frequency": "Yearly"

}

}

]

},

"status": "Success"

}

}

Description: for showing the complaince calendar Function : CalendarController.showComplianceCalendar()

* PATH: /getTasksForPerformerCalendar Method: GET

Response: JSON String

Description: gets all tasks assigned for calendar performer

wise

Function : CalendarController.getTasksForPerformer()

Status: Function Not Available

* PATH: /searchTasksForCalendar Method: POST

Response: JSON String

Description: searches the task for calendar

Function : CalendarController.searchTasksForCalendar()

Status: Function Not Available

* PATH: /getFrequencyFromTtrn Method: POST

Input: <http://localhost:9002/compliance/getFrequencyFromTtrn>

Response: JSON String

{

"response": {

"data": [

{

"frequency": "Yearly"

},

{

"frequency": "Five\_Yearly"

},

{

"frequency": "Half\_Yearly"

},

{

"frequency": "Quarterly"

},

{

"frequency": "One\_Time"

},

{

"frequency": "Monthly"

},

{

"frequency": "Three\_yearly"

},

{

"frequency": "Two\_yearly"

},

{

"frequency": "Four\_yearly"

},

{

"frequency": "Two\_Monthly"

},

{

"frequency": "Seven\_Yearly"

},

{

"frequency": "Four\_Monthly"

}

],

"status": "Success"

}

}

Description: gets assigned frequency

Function : CalendarController.getAssignedFrequency()

* PATH: /getentity Method: GET

Input: <http://localhost:9002/compliance/getentity>

Response: JSON String

{

"response": {

"data": [

{

"orga\_name": "Gennova Biopharmaceuticals Limited",

"orga\_id": 3

}

],

"status": "Success"

}

}

Description: gets entity

Function : CalendarController.getentity()

* PATH: /getunit Method: GET

Input: <http://localhost:9002/compliance/getunit?entity_id=3>

Response: JSON String

{

"response": {

"data": [

{

"loca\_id": 1,

"orga\_name": "Gennova Biopharmaceuticals Limited",

"loca\_name": "Hinjewadi EOU",

"orga\_id": 3

}

],

"status": "Success"

}

}

Description: gets unit

Function : CalendarController.getunit()

* PATH: /getFunction Method: GET

Input: <http://localhost:8082/getFunction/1>

Response: JSON String

{

     "response": {

         "data": {

             "function\_list": [

                 {

                     "loca\_id": 1,

                     "orga\_name": "Emcure Pharmaceuticals Limited",

                     "loca\_name": "Hinjewadi EOU",

                     "orga\_id": 2,

                     "dept\_name": "HR",

                     "dept\_id": 3

                 }

             ]

         },

         "status": "Success"

     }

}

Description: gets function

Function : CalendarController.getFunction()

* PATH: /getExeEvalFuncHeadList Method: POST

Input: <http://localhost:9002/compliance/getExeEvalFuncHeadList>

Response: JSON String

{

"response": {

"data": {

"OrganogramFilter": [

{

"Executor": [],

"FunctionHead": [],

"Evaluator": []

}

]

},

"status": "Success"

}

}

Description: gets executor, evaluator, function head list Function : CalendarController.getExeEvalFuncHeadList()

---------------------------------------------------------------------

---------------------------------------------------------------------

* PATH: /getentitylist Method: GET

Input: /compliance/getentitylist

Response: JSON String

{

"response": {

"data": {

"entity\_list": [

{

"orga\_name": "Emcure Pharmaceuticals Limited",

"orga\_id": 2

}

]

},

"status": "Success"

}

}

Description: gets the assigned entity Function : AssignTaskController.getentity()

* PATH: /getunitlist Method: GET

Input : /compliance/getunitlist?entity\_id=2

Response: JSON String

{

"response": {

"data": {

"unit\_list": [

{

"loca\_id": 6,

"orga\_name": "Emcure Pharmaceuticals Limited",

"loca\_name": "Pimpri F2",

"orga\_id": 2

},

{

"loca\_id": 11,

"orga\_name": "Emcure Pharmaceuticals Limited",

"loca\_name": "MO Hinjewadi",

"orga\_id": 2

},

{

"loca\_id": 10,

"orga\_name": "Emcure Pharmaceuticals Limited",

"loca\_name": "Emcure Pharma - Kurkumbh",

"orga\_id": 2

}

]

}

}

}

Description: gets unit list

Function : AssignTaskController.getunitlist()

* PATH: /getFunctionlist Method: POST

Input: /compliance/getFunctionlist

{

"orga\_id":1,

"loca\_list": [

{

"tmap\_loca\_id":1

}

]

}

Response: JSON String

{

"response": {

"data": {

"function\_list": [

{

"loca\_id": 1,

"orga\_name": "NA",

"loca\_name": "Hinjewadi EOU",

"orga\_id": 1,

"dept\_name": "HR",

"dept\_id": 3

}

]

},

"status": "Success"

}

}

Description: gets the assigned function list Function : AssignTaskController.getFunction()

* PATH: /getExecutorList Method: POST

Input: /compliance/getExecutorList

{

"orga\_id": 2,

"dept\_id": 3,

"loca\_list": [

{

"tmap\_loca\_id":25

}

]

}

Response: JSON String

{

"response": {

"data": {

"executor\_list": [

{

"user\_id": "1",

"user\_name": "Sagar Salvi"

},

{

"user\_id": "6",

"user\_name": "Manoj Pawar"

},

{

"user\_id": "61",

"user\_name": "Sanket Khatavkar"

},

{

"user\_id": "90",

"user\_name": "Basavaraj Miraje"

},

{

"user\_id": "273",

"user\_name": "Priyanka Patil"

},

{

"user\_id": "274",

"user\_name": "Paridhi Shivhare"

}

]

},

"status": "Success"

}

}

Description: gets the assigned executor list Function : AssignTaskController.getExecutorList()

* PATH: /getEvaluatorList Method: POST

Input: /compliance/getEvaluatorList

{

"orga\_id": 2,

"dept\_id": 3,

"loca\_list": [

{

"tmap\_loca\_id":25

}

]

}

Response: JSON String

{

"response": {

"data": {

"evaluator\_list": [

{

"user\_id": "1",

"user\_name": "Sagar Salvi"

},

{

"user\_id": "61",

"user\_name": "Sanket Khatavkar"

},

{

"user\_id": "321",

"user\_name": "Chetan Sharma"

},

{

"user\_id": "322",

"user\_name": "Saurabh Bansal"

}

]

},

"status": "Success"

}

}

Description: gets the assigned evaluator list Function : AssignTaskController.getEvaluatorList()

* PATH: /getFunHeadList Method: POST

Input: /compliance/getFunHeadList

{

"orga\_id": 2,

"dept\_id": 3,

"loca\_list": [{

"tmap\_loca\_id":25

}

]

}

Response: JSON String

{

"response": {

"data": {

"function\_head\_list": [

{

"user\_id": "1",

"user\_name": "Sagar Salvi"

},

{

"user\_id": "6",

"user\_name": "Manoj Pawar"

},

{

"user\_id": "321",

"user\_name": "Chetan Sharma"

},

{

"user\_id": "322",

"user\_name": "Saurabh Bansal"

}

]

},

"status": "Success"

}

}

Description: gets the assigned function head list Function : AssignTaskController.getFunHeadList()

* PATH: /getTaskListToAssign Method: POST

Input: compliance/getTaskListToAssign?pagination=true&page\_id=1&per\_page=10

{

"country\_id":1,

"state\_id":2,

"cat\_id":7,

"legi\_id":58,

"rule\_id":1059,

"orga\_id":1,

"loca\_id":1

}

Response: JSON String

{

"response": {

"data": {

"task\_list\_to\_assign": {}

},

"status": "Success"

}

}

Description: gets taks for assigning operation Function : AssignTaskController. ()

* PATH: /getExeListForChangeOwner Method: POST

Input: /getExeListForChangeOwner?orga\_id=2&loca\_id=1&dept\_id=3

Response: JSON String

{

     "response": {

         "data": {

             "Executor": [

                 {

                     "user\_id": "1",

                     "user\_name": "Prashant1 Kolage1"

                 }

             ]

         },

         "status": "Success"

      }

}

Description: gets executor list for changing the owner Function : AssignTaskController.getExeListForChangeOwner()

* PATH: /getEvalListForChangeOwner Method: POST

Input: /getExeListForChangeOwner?orga\_id=2&loca\_id=1&dept\_id=3

Response: JSON String

{

     "response": {

         "data": {

             "Executor": [

                 {

                     "user\_id": "1",

                     "user\_name": "Prashant1 Kolage1"

                 }

             ]

         },

         "status": "Success"

      }

}

Description: gets evaluator list for changing the owner Function : AssignTaskController.getEvalListForChangeOwner()

* PATH: /getFunHeadListForChangeOwner Method: POST

Input: /getFunHeadListForChangeOwner?orga\_id=2&loca\_id=1&dept\_id=3

Response: JSON String

{

     "response": {

         "data": {

             "Function\_Head": []

         },

         "status": "Success"

     }

}

Description: gets function head list for changing the owner Function : AssignTaskController.getFunHeadListForChangeOwner()

* PATH: /searchcomplianceownerpage Method: POST

Input: <http://localhost:9002/compliance/searchcomplianceownerpage>

{

    "orga\_id":"",

    "loca\_id":"",

    "dept\_id":"",

    "pr\_user\_id":"",

    "rw\_user\_id":"",

    "fh\_user\_id":"",

    "legi\_id":"",

    "rule\_id":""

}

Response: JSON String

{

    "response": {

        "data": {

            "repoData": [

                {

                    "tmap\_id": 4,

                    "tmap\_client\_tasks\_id": "D02030000003",

                    "task\_legi\_name": "The Companies Act, 2013",

                    "task\_rule\_name": "The Companies (Audit and Auditors) Rules, 2014",

                    "task\_reference": "Section 143(9)",

                    "task\_activity\_who": "Company",

                    "task\_activity\_when": "On the applicability of the Act",

                    "task\_activity": "Ensure that the auditor complies with all the applicable accounting standards",

                    "task\_procedure": "",

                    "tmap\_enable\_status": "1",

                    "executor\_id": "1",

                    "executor\_name": "Prashant1 Kolage1",

                    "evaluator\_id": "1",

                    "orga\_id": "2",

                    "loca\_id": "2",

                    "dept\_id": "3",

                    "task\_lexcare\_id": "C001112111",

                    "function\_head\_id": "1",

                    "task\_legi\_id": "135",

                    "task\_rule\_id": "830"

                }

            ]

        },

        "status": "Success"

    }

}

Description: search complaince owner page

Function : AssignTaskController.searchcomplianceownerpage()

* PATH: /uploadAssignTaskList Method: POST

Response: JSON String

Description: upload assign task list

Function : AssignTaskController.uploadAssignTaskList()

Status :Function not Available

* PATH: /searchActivationPage Method: POST

Input: <http://127.0.0.1:8082/searchActivationPage>

Response: JSON String

{

     "response": {

         "data": {

             "all\_tasks": []

         },

         "status": "Success"

      }

}

Description: for getting the activate page

Function : AssignTaskController.searchActivationPage()

* PATH: /getExeListForActivationPage Method: POST

Input: /getExeListForActivationPage?orga\_id=2&loca\_id=1&dept\_id=3

Response: JSON String

{

     "response": {

         "data": {

             "Executor": []

         },

         "status": "Success"

     }

}

Description: get the executor list for activation page Function : AssignTaskController.getExeListForActivationPage()

* PATH: /getEvalListForActivationPage Method: POST

Inxput : /getEvalListForActivationPage?orga\_id=2&loca\_id=1&dept\_id=3

Response: JSON String

{

     "response": {

         "data": {

             "Evaluator": []

         },

         "status": "Success"

     }

}

Description: get the evaluator list for activation page Function : AssignTaskController.getEvalListForActivationPage()

* PATH: /searchEnableDisablePage Method: POST

Input: <http://localhost:9002/compliance/searchEnableDisablePage>

{

    "orga\_id": "2",

    "loca\_id": "1",

    "dept\_id": "3",

    "pr\_user\_id": "1",

    "rw\_user\_id":"1",

    "legi\_id":"3",

    "rule\_id": "4"

}

Response: JSON String

{

    "response": {

        "data": {

            "all\_tasks": [

                {

                    "task\_legi\_id": 135,

                    "task\_rule\_name": "The Companies (Audit and Auditors) Rules, 2014",

                    "executor\_name": "Prashant1 Kolage1",

                    "task\_activity": "Ensure that the auditor complies with all the applicable accounting standards",

                    "orga\_id": 2,

                    "task\_legi\_name": "The Companies Act, 2013",

                    "tmap\_enable\_status": "1",

                    "tmap\_id": 4,

                    "tmap\_client\_tasks\_id": "D02030000003",

                    "task\_activity\_when": "On the applicability of the Act",

                    "loca\_id": 2,

                    "function\_head\_id": 1,

                    "task\_activity\_who": "Company",

                    "task\_reference": "Section 143(9)",

                    "evaluator\_id": 1,

                    "task\_lexcare\_id": "C001112111",

                    "task\_rule\_id": 830,

                    "task\_procedure": "",

                    "dept\_id": 3,

                    "executor\_id": 1

                }

            ]

        },

        "status": "Success"

    }

}

Description: search for getting the enable or disable page Function : AssignTaskController.searchEnableDisablePage()

* PATH: / fetchreports

Method: POST

Input : <http://localhost:9002/compliance/fetchreports>

{

"ChooseROrL":"reportPeriod",

"legal\_status":"NA",

"entity\_id":0,

"unit\_id":0,

"func\_id":0,

"exec\_id":0,

"eval\_id":0,

"from\_date":"01-05-2021",

"to\_date":"25-05-2021",

"task\_impact":"NA"

}

Response: JSON String

{

"response": {

"data": {

"reportSubTaskList": [],

"reportList": [

{

"task\_rule\_name": "The Companies (Acceptance of Deposits) Rules, 2014",

"task\_impact": "Low",

"ttrn\_legal\_due\_date": "12-05-2021",

"task\_status": "Non Complied",

"entity\_name": "Zuventus Healthcare Limited",

"executor\_name": "Sheetal Kulkarni",

"task\_activity": "Obtain credit rating for deposits accepted by company in the manner prescribed once in a year and copy of the rating shall sent to the Registrar along with return of deposit in form DPT-3",

"ttrn\_reason\_for\_non\_compliance": "",

"task\_legi\_name": "The Companies Act, 2013",

"unit\_name": "Zuventus Healthcare Secretarial",

"task\_reference": "Section 73(2) and Rule 3(8)",

"ttrn\_submitted\_date": "",

"task\_who": "Public Company having: \n(i) Net worth of not less than Rs. 100 crore \n(ii) Turnover of not less than Rs. 500 crores",

"document\_attached": 0,

"tmap\_client\_task\_id": "D35240001423",

"ttrn\_performer\_comments": "",

"function\_name": "Secretarial",

"document\_list": [],

"task\_when": "In case accepting deposits from public",

"ttrn\_completed\_date": "",

"task\_frequency": "Yearly"

},

{

"task\_rule\_name": "The Companies (Acceptance of Deposits) Rules, 2014",

"task\_impact": "Low",

"ttrn\_legal\_due\_date": "12-05-2021",

"task\_status": "Non Complied",

"entity\_name": "Gennova Biopharmaceuticals Limited",

"executor\_name": "Rutuja Gohad",

"task\_activity": "Obtain credit rating for deposits accepted by company in the manner prescribed once in a year and copy of the rating shall sent to the Registrar along with return of deposit in form DPT-3",

"ttrn\_reason\_for\_non\_compliance": "",

"task\_legi\_name": "The Companies Act, 2013",

"unit\_name": "Gennova Biopharma - Secreterial",

"task\_reference": "Section 73(2) and Rule 3(8)",

"ttrn\_submitted\_date": "",

"task\_who": "Public Company having: \n(i) Net worth of not less than Rs. 100 crore \n(ii) Turnover of not less than Rs. 500 crores",

"document\_attached": 0,

"tmap\_client\_task\_id": "D37240002112",

"ttrn\_performer\_comments": "",

"function\_name": "Secretarial",

"document\_list": [],

"task\_when": "In case accepting deposits from public",

"ttrn\_completed\_date": "",

"task\_frequency": "Yearly"

}

]

},

"status": "Success"

}

}

Description: get the evaluator list for activation page Function : AssignTaskController.getEvalListForActivationPage()