Backend API:

# Routes

# This file defines all application routes (Higher priority routes first)

# ~~~~

# Home page - we use the @controllers syntax here so that the router knows we are referring to a class. In turn it

# will call on the GlobalSettings.getController method to resolve the actual controller instance.

# GET / @controllers.Application.index

# Climate Service

GET /climateService/getClimateService/:name/json @controllers.ClimateServiceController.getClimateService(name: String, format: String="json")

GET /climateService/getClimateService/id/:id @controllers.ClimateServiceController.getClimateServiceById(id: Long)

GET /climateService/getAllClimateServices/json @controllers.ClimateServiceController.getAllClimateServices(format: String="json")

GET /climateService/getAllMostRecentClimateServicesByCreateTime/json @controllers.ClimateServiceController.getAllClimateServicesOrderByCreateTime(format: String="json")

GET /climateService/getAllMostRecentClimateServicesByLatestAccessTime/json @controllers.ClimateServiceController.getAllClimateServicesOrderByLatestAccessTime(format: String="json")

GET /climateService/getAllMostUsedClimateServices/json @controllers.ClimateServiceController.getAllClimateServicesOrderByCount(format: String="json")

POST /climateService/addClimateService @controllers.ClimateServiceController.addClimateService

GET /climateService/getAllServiceEntries/json @controllers.ClimateServiceController.getAllServiceEntries(format: String="json")

POST /climateService/addServiceEntry @controllers.ClimateServiceController.addServiceEntry

PUT /climateService/updateClimateService/id/:id @controllers.ClimateServiceController.updateClimateServiceById(id: Long)

PUT /climateService/updateClimateService/name/:oldName @controllers.ClimateServiceController.updateClimateServiceByName(oldName: String)

DELETE /climateService/deleteClimateService/id/:id @controllers.ClimateServiceController.deleteClimateServiceById(id: Long)

DELETE /climateService/deleteClimateService/name/:name @controllers.ClimateServiceController.deleteClimateServiceByName(name: String)

# Instrument

GET /instrument/getAllInstruments/json @controllers.InstrumentController.getAllInstruments(format: String="json")

GET /instrument/getInstrument/id/:id/json @controllers.InstrumentController.getInstrument(id: Long, format: String="json")

POST /instrument/addInstrument @controllers.InstrumentController.addInstrument

PUT /instrument/updateInstrument/id/:id @controllers.InstrumentController.updateInstrumentById(id: Long)

DELETE /instrument/deleteInstrument/id/:id @controllers.InstrumentController.deleteInstrument(id: Long)

# Dataset

GET /dataset/getAllDatasets/json @controllers.DatasetController.getAllDatasets(format: String="json")

GET /dataset/getDataset/id/:id/json @controllers.DatasetController.getDataset(id: Long, format: String="json")

POST /dataset/addDataset @controllers.DatasetController.addDataset

PUT /dataset/updateDataset/id/:id @controllers.DatasetController.updateDatasetById(id: Long)

DELETE /dataset/deleteDataset/id/:id @controllers.DatasetController.deleteDataset(id: Long)

POST /dataset/queryDataset @controllers.DatasetController.queryDatasets

# Parameter

GET /parameter/getAllParameters/json @controllers.ParameterController.getAllParameters(format: String="json")

GET /parameter/getParameter/service/:id/name/:name/json @controllers.ParameterController.getParameterByName(id:Long, name: String, format: String="json")

GET /parameter/getParameter/id/:id/json @controllers.ParameterController.getParameterById(id: Long, format: String="json")

POST /parameter/addParameter @controllers.ParameterController.addParameter

PUT /parameter/updateParameter/id/:id @controllers.ParameterController.updateParameterById(id: Long)

PUT /parameter/updateParameter/name/:oldName @controllers.ParameterController.updateParameterByName(oldName: String)

DELETE /parameter/deleteParameter/service/:id/name/:name @controllers.ParameterController.deleteParameterByName(id:Long, name: String)

# Users

DELETE /users/delete/:id @controllers.UserController.deleteUser(id: Long)

GET /users/getAllUsers/json @controllers.UserController.getAllUsers(format: String="json")

DELETE /users/delete/userName/:userName/password/:password @controllers.UserController.deleteUserByUserNameandPassword(userName: String, password: String)

POST /users/register @controllers.UserController.userRegister

POST /users/login @controllers.UserController.userLogin

GET /users/search/:display\_name/json @controllers.UserController.userSearch(display\_name: String, format: String="json")

PUT /users/setprofile/:id @controllers.UserController.setProfile(id: Long)

GET /users/getprofile/:id/json @controllers.UserController.getProfile(id: Long, format: String="json")

GET /users/follow/followerId/:followerId/followeeId/:followeeId @controllers.UserController.userFollow(followerId: Long, followeeId: Long)

GET /users/unfollow/followerId/:followerId/followeeId/:followeeId @controllers.UserController.userUnfollow(followerId: Long, followeeId: Long)

GET /users/getFollowers/:id @controllers.UserController.getFollowers(id: Long)

GET /users/getFollowees/:id @controllers.UserController.getFollowees(id: Long)

GET /users/sendFriendRequest/sender/:senderId/receiver/:receiverId @controllers.UserController.sendFriendRequest(senderId: Long, receiverId: Long)

GET /users/getFriendRequests/userId/:userId @controllers.UserController.getFriendRequests(userId: Long)

GET /users/acceptFriendRequest/userId/:userId/sender/:senderId @controllers.UserController.acceptFriendRequest(userId: Long, senderId: Long)

GET /users/rejectFriendRequest/userId/:userId/sender/:senderId @controllers.UserController.rejectFriendRequest(userId: Long, senderId: Long)

GET /users/getFriends/userId/:userId @controllers.UserController.getFriends(userId: Long)

GET /users/deleteFriend/userId/:userId/friendId/:friendId @controllers.UserController.deleteFriend(userId: Long, friendId: Long)

# Workflow

POST /workflow/post @controllers.WorkflowController.post

POST /workflow/updateWorkflow @controllers.WorkflowController.updateWorkflow

POST /workflow/deleteWorkflow @controllers.WorkflowController.deleteWorkflow

GET /workflow/get/workflowID/:wfID/userID/:userID/json @controllers.WorkflowController.get(wfID: Long, userID: Long, format: String="json")

GET /workflow/getTimeline/:id/offset/:offset/json @controllers.WorkflowController.getTimeLine(id: Long, offset: Long, format: String="json")

GET /workflow/getWorkflowList/:id/json @controllers.WorkflowController.getWorkflowList(id: Long, format: String="json")

POST /workflow/uploadImage/:id @controllers.WorkflowController.uploadImage(id:Long)

GET /workflow/getPublicWorkflow/json @controllers.WorkflowController.getPublicWorkflow(format: String="json")

GET /workflow/getTags/workflowId/:wfId @controllers.WorkflowController.getTags(wfId: Long)

POST /workflow/setTag @controllers.WorkflowController.setTag

GET /workflow/deleteTag/workflowId/:wfId/tag/:tag @controllers.WorkflowController.deleteTag(wfId: Long, tag: String)

GET /workflow/getByTag/tag/:tag @controllers.WorkflowController.getByTag(tag: String)

GET /workflow/getByTitle/title/:title @controllers.WorkflowController.getByTitle(title: String)

GET /workflow/getTop3WorkFlow @controllers.WorkflowController.getTop3WorkFlow()

# Mail

POST /mail/sendMail @controllers.MailController.sendMail

GET /mail/getMailDetail/mailId/:mailId @controllers.MailController.readMail(mailId: Long)

GET /mail/getInbox/:id/json @controllers.MailController.getInbox(id: Long, format: String="json")

GET /mail/getOutbox/:id/json @controllers.MailController.getOutbox(id: Long, format: String="json")

#Suggestion

POST /suggestion/publishSuggestion @controllers.SuggestionsController.publishSuggestion

POST /suggestion/addTag @controllers.SuggestionsController.addTag

GET /suggestion/voteToSuggestion/:suggestionId @controllers.SuggestionsController.voteToSuggestion(suggestionId: Long)

GET /suggestion/getSuggestionForWorkflow/:workflowId @controllers.SuggestionsController.getSuggestionForWorkflow(workflowId: Long)

# Tag

#input:String tag

POST /tag/createTag @controllers.TagController.createTag

POST /tag/deleteTag @controllers.TagController.deleteTag

#Group

POST /group/createGroup @controllers.GroupUsersController.createGroup

POST /group/addMembersToGroup @controllers.GroupUsersController.addMembersToGroup

GET /group/getGroupList/:userID/json @controllers.GroupUsersController.getGroupList(userID: Long, format: String="json")

GET /group/getGroupMember/:groupId/json @controllers.GroupUsersController.getGroupMember(groupId: Long, format: String="json")

# Map static resources from the /public folder to the /assets URL path

GET /assets/\*file controllers.Assets.at(path="/", file)

#Workflow

#GET /workflow/getTimeline/:id/offset/:offset/json @controllers.UserController.userSearch(id: Long, offset: Long, format: String="json")

POST /workflow/addComment @controllers.WorkflowController.addComment

GET /workflow/getComments/:workflowId @controllers.WorkflowController.getComments(workflowId: Long)

#Comment

POST /Comment/addReply @controllers.CommentController.addReply

GET /Comment/getReply/:commentId @controllers.CommentController.getReply(commentId: Long)

GET /Comment/getReply/:commentId @controllers.CommentController.getReply(commentId: Long)

GET /Comment/thumbUp/:commentId @controllers.CommentController.thumbUp(commentId: Long)

GET /Comment/thumbDown/:commentId @controllers.CommentController.thumbDown(commentId: Long)

#######################################################################################################################################

Frontend APIs:

# Routes

# This file defines all application routes (Higher priority routes first)

# ~~~~

# Home page

GET /climatehome controllers.ClimateServiceController.home(email ?= null, vfile ?= null, dataset ?= null)

# GET / controllers.ClimateServiceController.home(email ?= null, vfile ?= null, dataset ?= null)

# BugReporting

GET /bugs controllers.BugReportController.reports()

GET /newReport controllers.BugReportController.newReport()

GET /bug/list controllers.BugReportController.list()

POST /delete/report controllers.BugReportController.deleteReport()

POST /solve/report controllers.BugReportController.solveReport()

# About Us

GET /aboutus controllers.AboutusController.aboutUs()

GET /aboutProject controllers.AboutusController.aboutProject()

# Climate Model

GET /climate controllers.ClimateServiceController.home(email ?= null, vfile ?= null, dataset ?= null)

GET /climate/tutorial controllers.ClimateServiceController.tutorial()

GET /climate/climateServices controllers.ClimateServiceController.climateServices()

GET /climate/mostRecentlyAddedClimateServices controllers.ClimateServiceController.mostRecentlyAddedClimateServices()

GET /climate/mostRecentlyUsedClimateServices controllers.ClimateServiceController.mostRecentlyUsedClimateServices()

GET /climate/mostPopularClimateServices controllers.ClimateServiceController.mostPopularClimateServices()

POST /climate/new/climateServices controllers.ClimateServiceController.newClimateService()

POST /climate/delete/climateServices controllers.ClimateServiceController.deleteClimateService()

POST /climate/download/climateServices controllers.ClimateServiceController.downloadClimateService()

POST /climate/edit/climateServices controllers.ClimateServiceController.editClimateService()

GET /climate/new/climateServices controllers.ClimateServiceController.addClimateServices()

GET /climate/oneService controllers.ClimateServiceController.oneService(url ?= null)

#Keyword search

GET /climate/datasets controllers.DatasetController.datasetList()

GET /climate/searchDataSet controllers.DatasetController.searchDataset()

GET /climate/getSearchResult controllers.DatasetController.getSearchResult()

#Map static resources from the /public folder to the /assets URL path

GET /assets/\*file controllers.Assets.at(path="/public", file)

#Workflow project start here

GET / controllers.Application.home()

GET /home controllers.Application.home()

GET /login controllers.Application.login()

POST /login controllers.Application.authenticate()

GET /index controllers.Application.index()

GET /logout controllers.Application.logout()

GET /profile/:id controllers.ProfileController.profile(id: Long)

GET /signup controllers.SignupController.signUp()

POST /signup/register controllers.SignupController.register()

GET /search controllers.SearchController.index()

GET /searchResult controllers.SearchController.search(category: String ?= "", keywd: String ?= "")

GET /follow/:id controllers.ProfileController.follow(id: Long)

GET /unfollow/:id controllers.ProfileController.unfollow(id: Long)

GET /befriend/:id controllers.ProfileController.sendRequest(id: Long)

GET /deletefriend/:id controllers.ProfileController.deleteFriend(id: Long)

# display and show work flow

GET /workflow controllers.WorkflowController.main()

GET /timeline/:offset controllers.TimelineController.main(offset: Long)

POST /workflow/new controllers.WorkflowController.createFlow()

GET /workflow/getPublicWorkflow controllers.WorkflowController.getPublicWorkflow()

GET /workflow/get/:wid controllers.WorkflowController.workflowDetail(wid: Long)

GET /workflow/update/:wid controllers.WorkflowController.edit(wid: Long)

POST /workflow/edit/:wid controllers.WorkflowController.editFlow(wid: Long)

POST /workflow/addComment/:wid controllers.WorkflowController.addComment(wid: Long)

POST /workflow/addReply/:toUserId/:commentId/:wid controllers.WorkflowController.addReply(toUserId: Long, commentId: Long, wid: Long)

GET /workflow/thumbUp/:commentId/:wid controllers.WorkflowController.thumbUp(commentId: Long, wid: Long)

GET /workflow/thumbDown/:commentId/:wid controllers.WorkflowController.thumbDown(commentId: Long, wid: Long)

GET /workflow/delete/:wid controllers.WorkflowController.deleteWorkflow(wid: Long)

GET /workflow/tags/add/wf/:wid/tag/:tg controllers.WorkflowController.addTag(wid: Long, tg: String)

GET /workflow/tags/delete/wf/:wid/tag/:tg controllers.WorkflowController.deleteTag(wid: Long, tg: String)

GET /workflow/import controllers.WorkflowController.parseXML(wid: Long)

# group

GET /group/create controllers.GroupController.create()

POST /group/creategroup controllers.GroupController.createGroup()

GET /group/join controllers.GroupController.join()

POST /group/joingroup controllers.GroupController.joinGroup()

# notifications

GET /notifications controllers.NotificationController.main()

GET /friends/accept/:id controllers.NotificationController.accpetFriend(id: Long)

GET /friends/reject/:id controllers.NotificationController.rejectFriend(id: Long)

POST /mail/sendmail controllers.NotificationController.sendMessage()

GET /mail/detail/:id controllers.NotificationController.pmdetailpage(id: Long)

GET /notifications/count controllers.NotificationController.getNotifications()

# suggestion

POST /suggestion/addSuggestion/:wid controllers.WorkflowController.addSuggestion(wid: Long)

POST /suggestion/addSuggestionTag/:suggestionID controllers.WorkflowController.addSuggestionTag(suggestionID: Long)

GET /suggestion/voteToSuggestion/:suggestionID controllers.WorkflowController.voteToSuggestion(suggestionID: Long)