

/v3/api-docs

OAS3

Explore

OpenAPI definition

/v3/api-docs

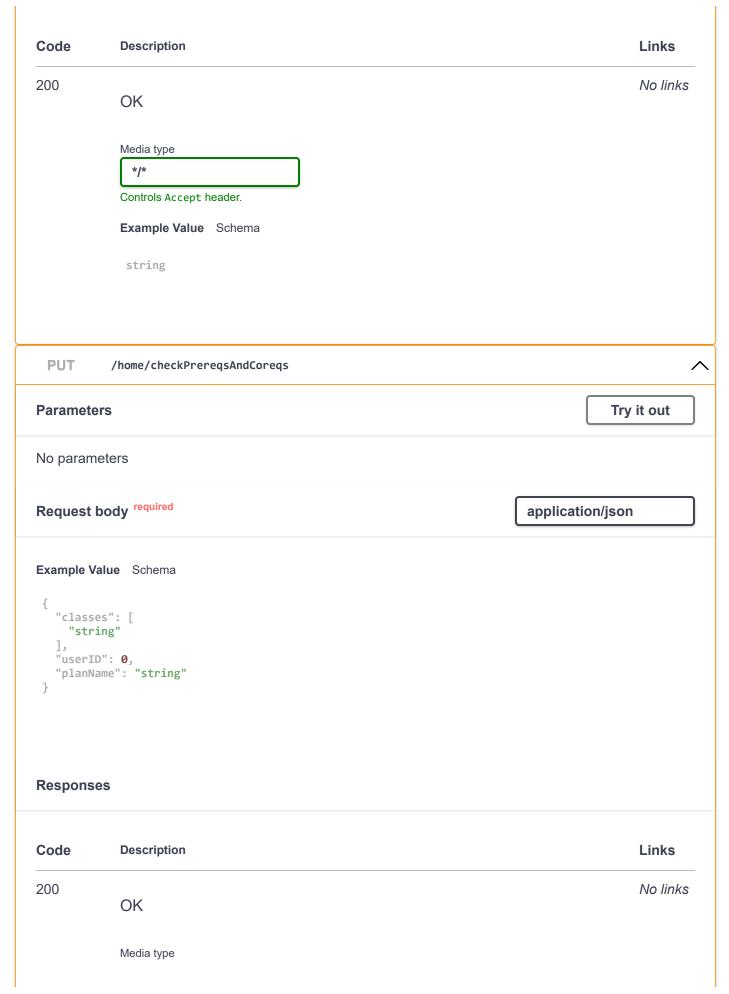
Servers

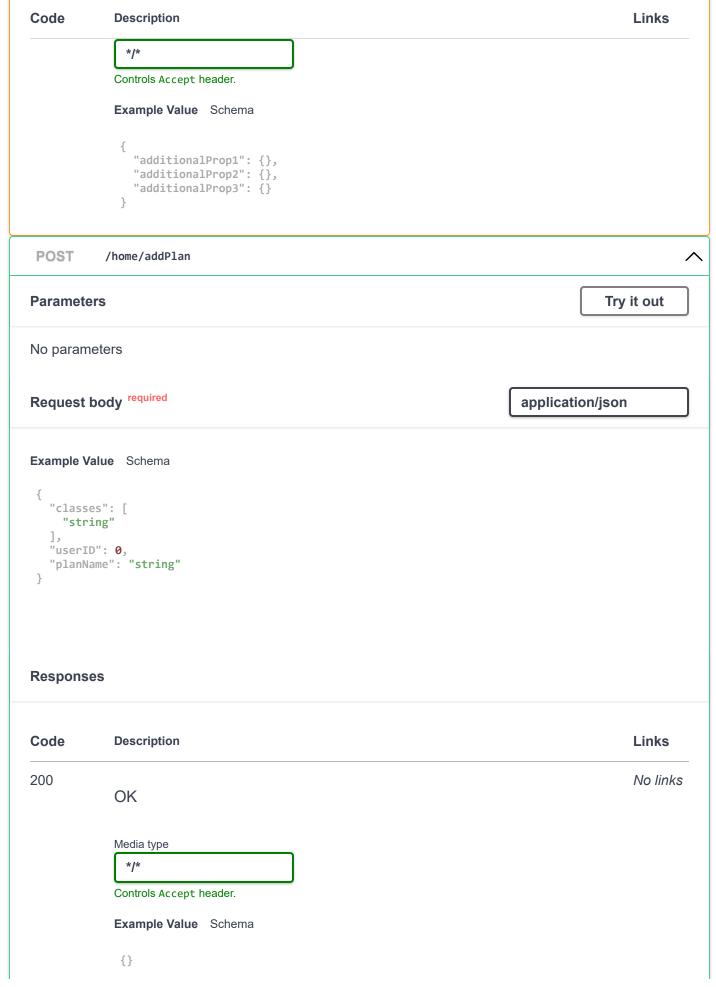
http://localhost:8060 - Generated server url

HomeApi Controller Class that has all endpoints relating to the homepage including most stuff related to the creation & editing of student plans



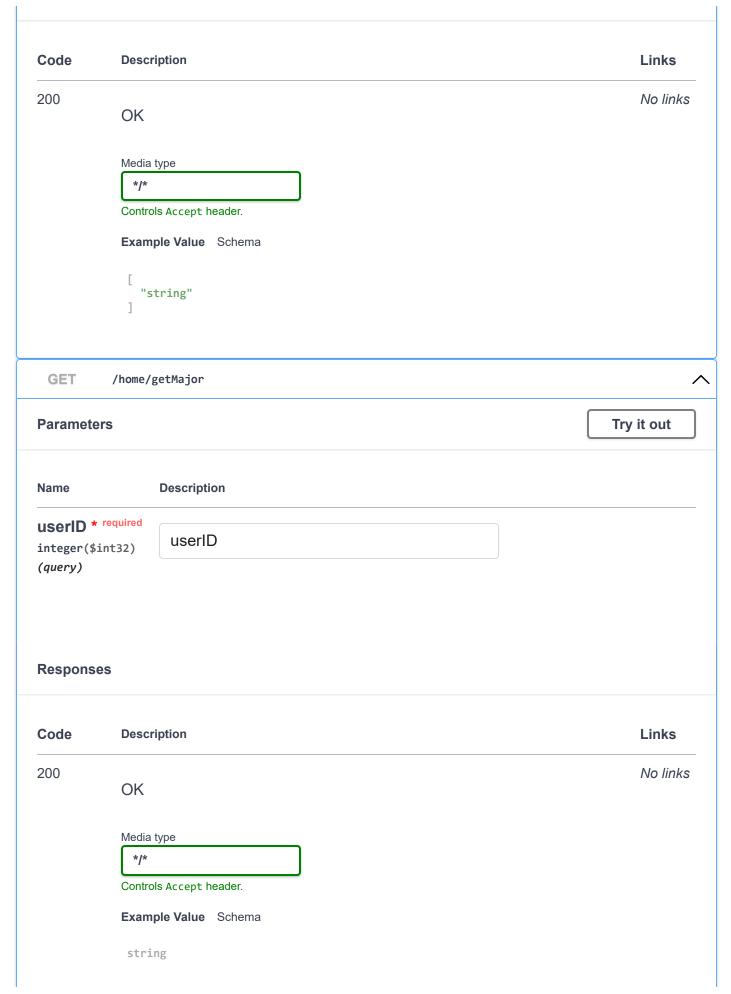
```
PUT
           /home/updatePlan
Parameters
                                                                                       Try it out
No parameters
Request body required
                                                                          application/json
Example Value Schema
   "classes": [
    "string"
   "userID": 0,
   "planName": "string"
Responses
```

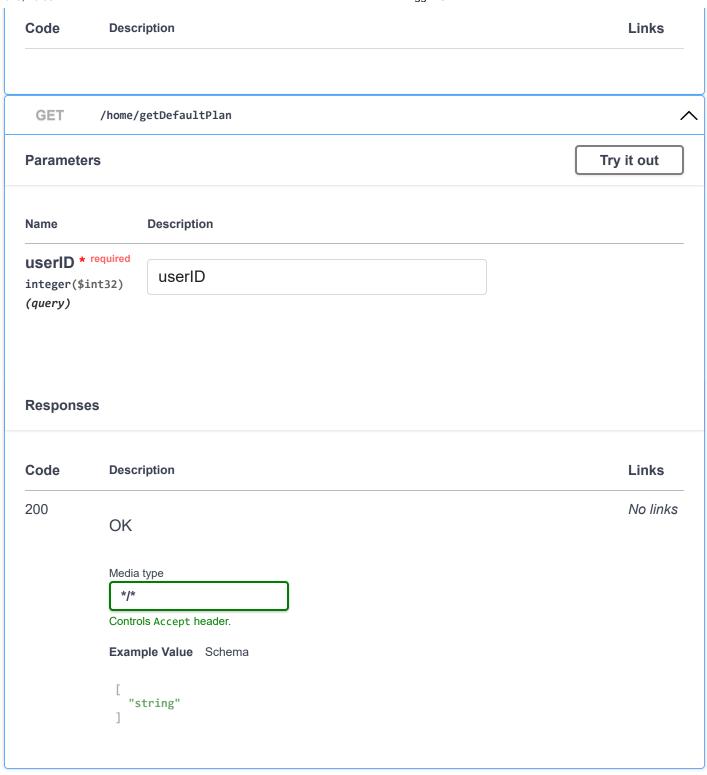


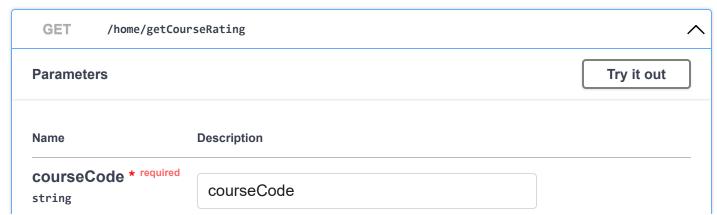


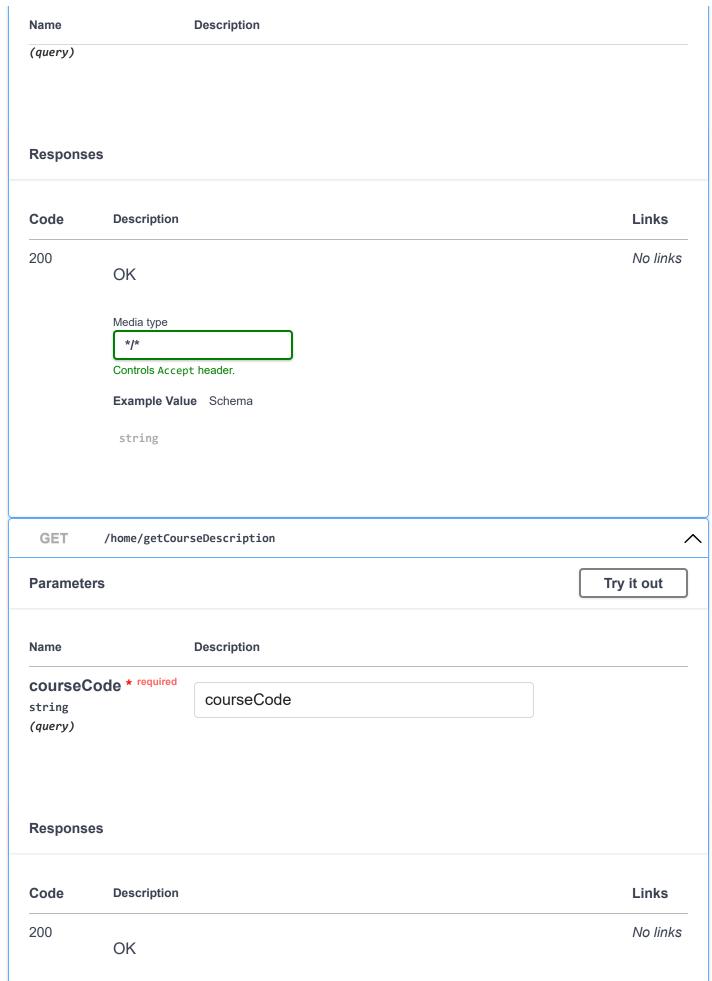
POST Parameter	/home/addCourse	Comment	
		Comment	,
Parameter			
	'S		Try it out
Name		Description	
commentInput * required string (query) courseCode * required string		commentInput	
		courseCode	
(query)			
Response	s		
Code	Description		Links
200	OK		No links
	Media type		
	/		
	Controls Accept h		
	Example Value	Scriema	
	string		
GET	/home		,

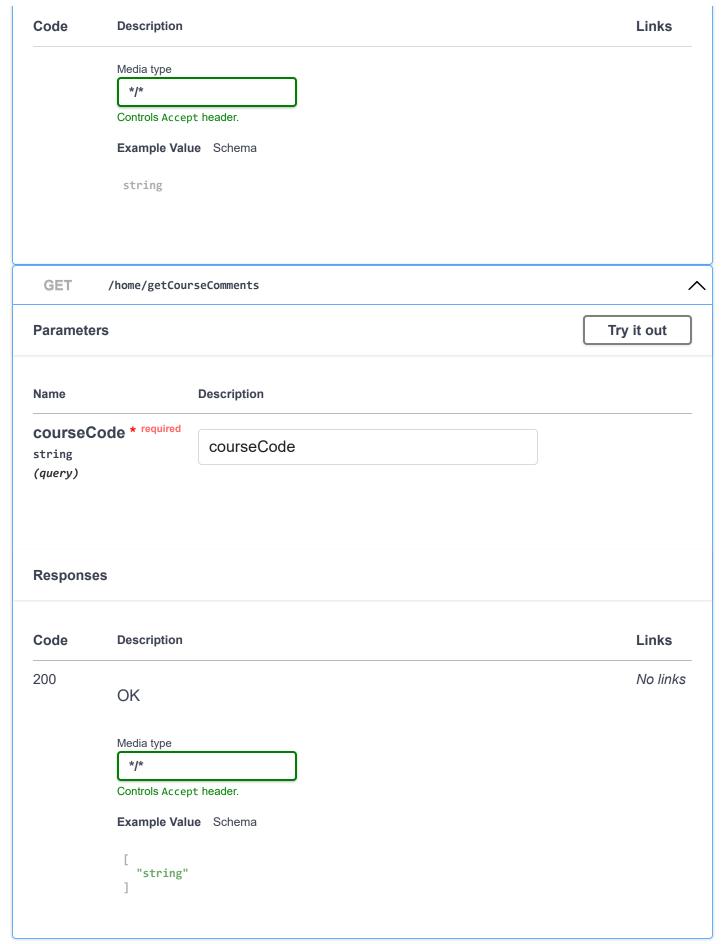
Name	Description	
userID * required integer(\$int32) (query)	userID	
olanName * ^{requi} string (query)	planName	
Responses		
Code Descr	ption	Links
000 OK		No links
Exam	s Accept header. Sle Value Schema	
GET /home/g	etSchedules	
Parameters		Try it out
lame	Description	
userID * required integer(\$int32) (query)	userID	
Responses		

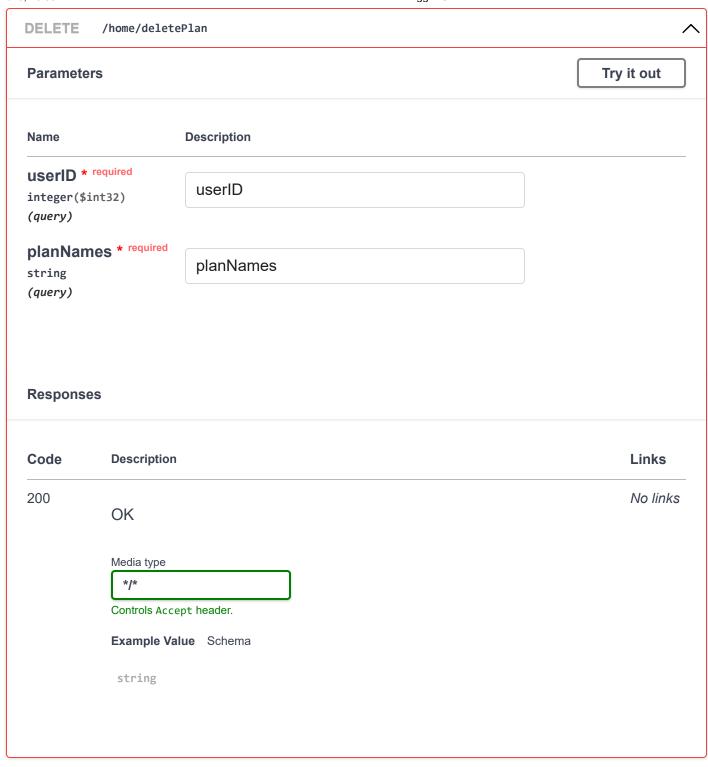










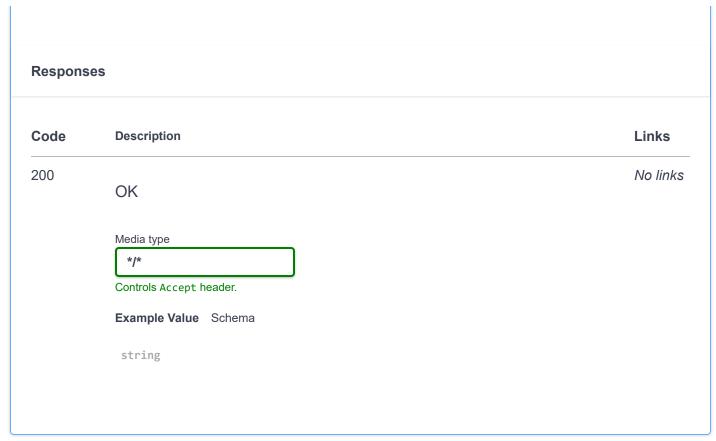


Test Controller a controller that contains a test endpoint we use to tell if our app's backend is running properly

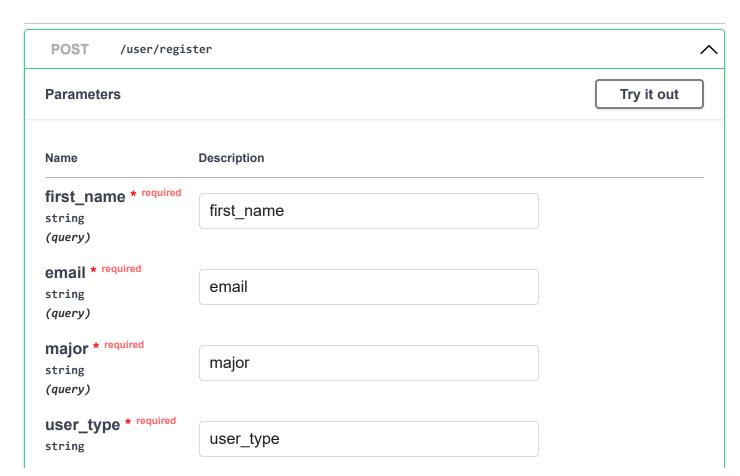
Parameters

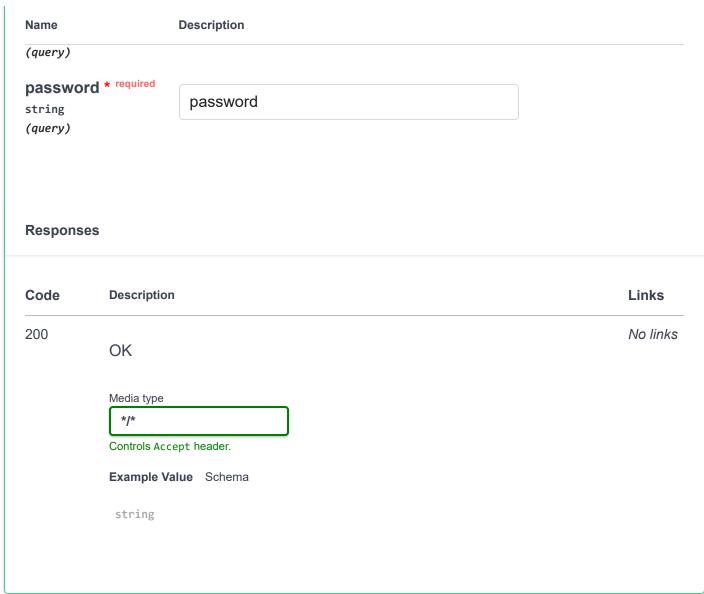
Try it out

No parameters



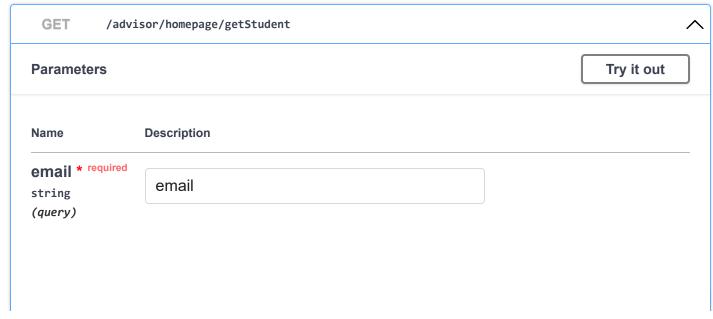
Register Controller to manage user registration endpoints including registering a new user into the database and loading there information

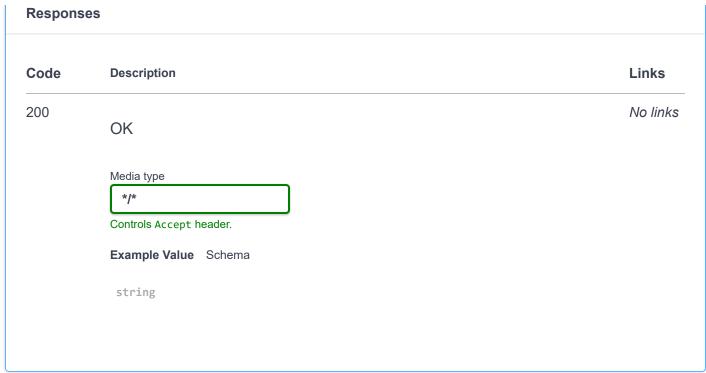




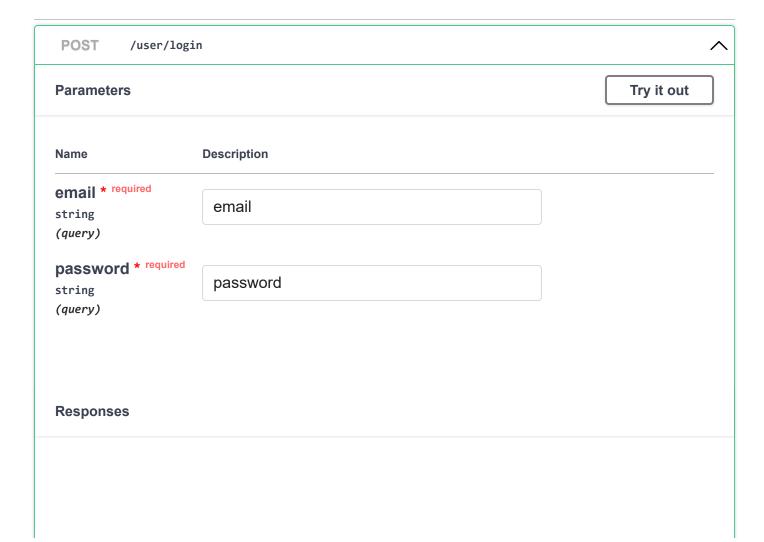
AdvisorHomePage Controller Contains all endpoints related to getting and setting advisor home page information

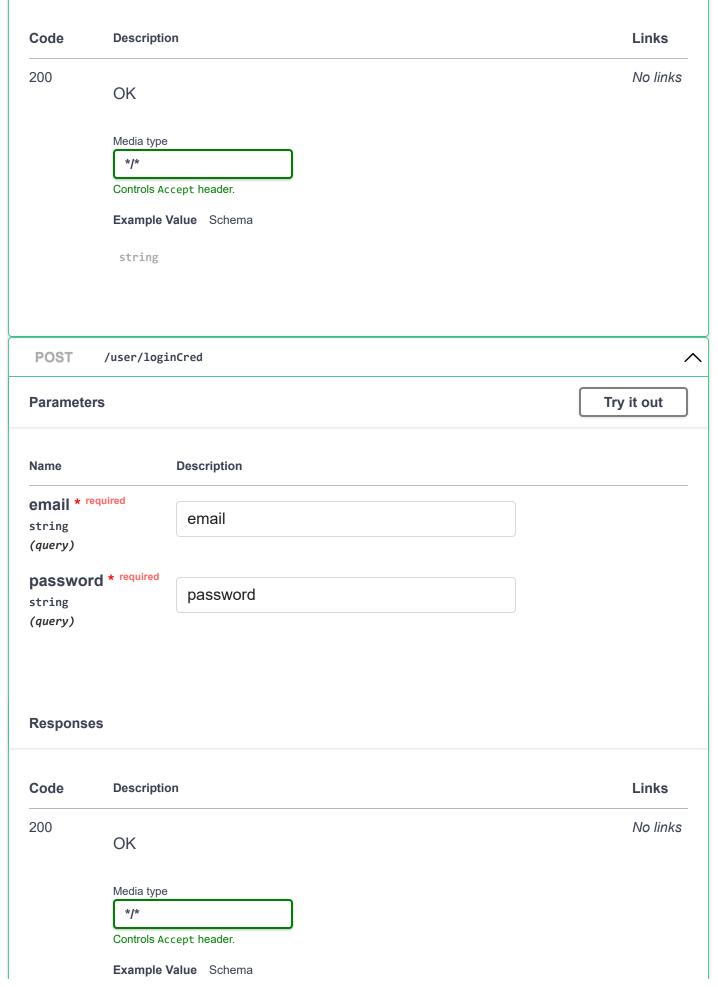


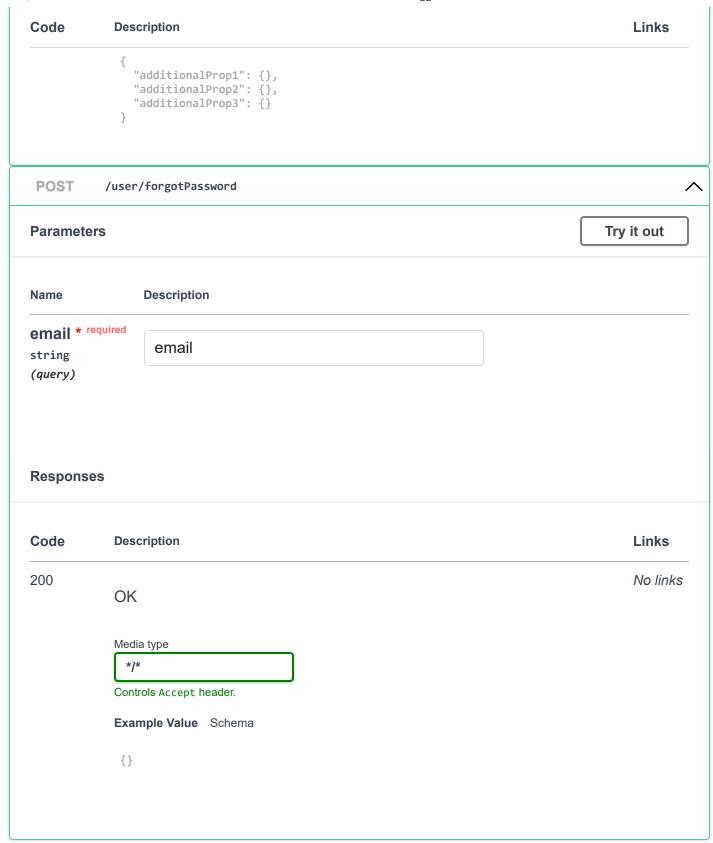




Login Controller simple user login endpoint for testing and complex login to return credentials for shared preferences

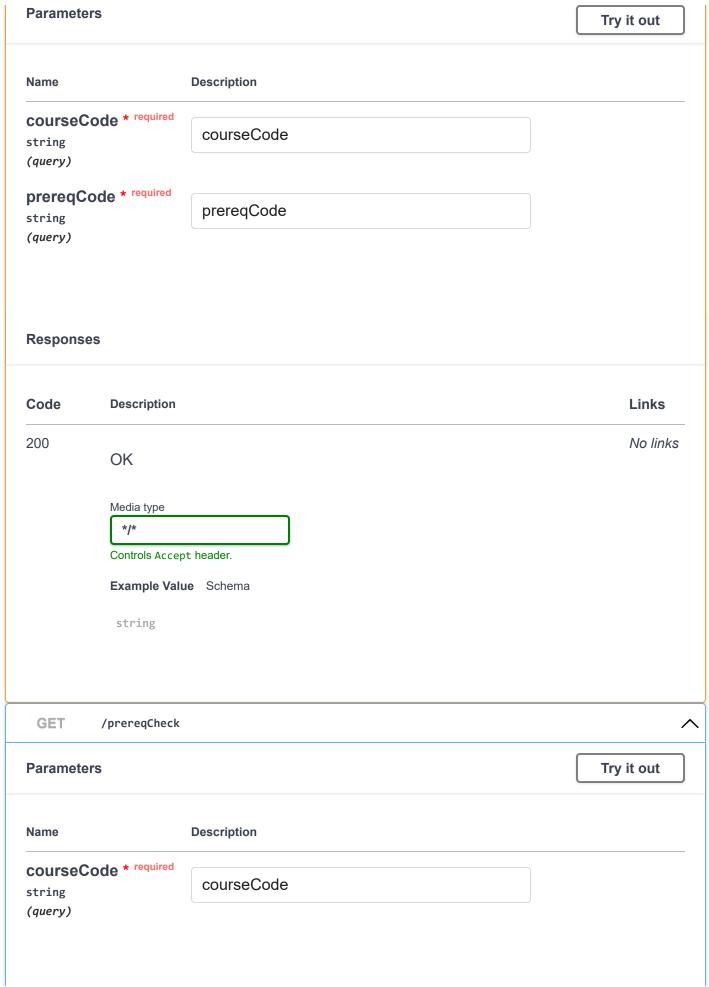


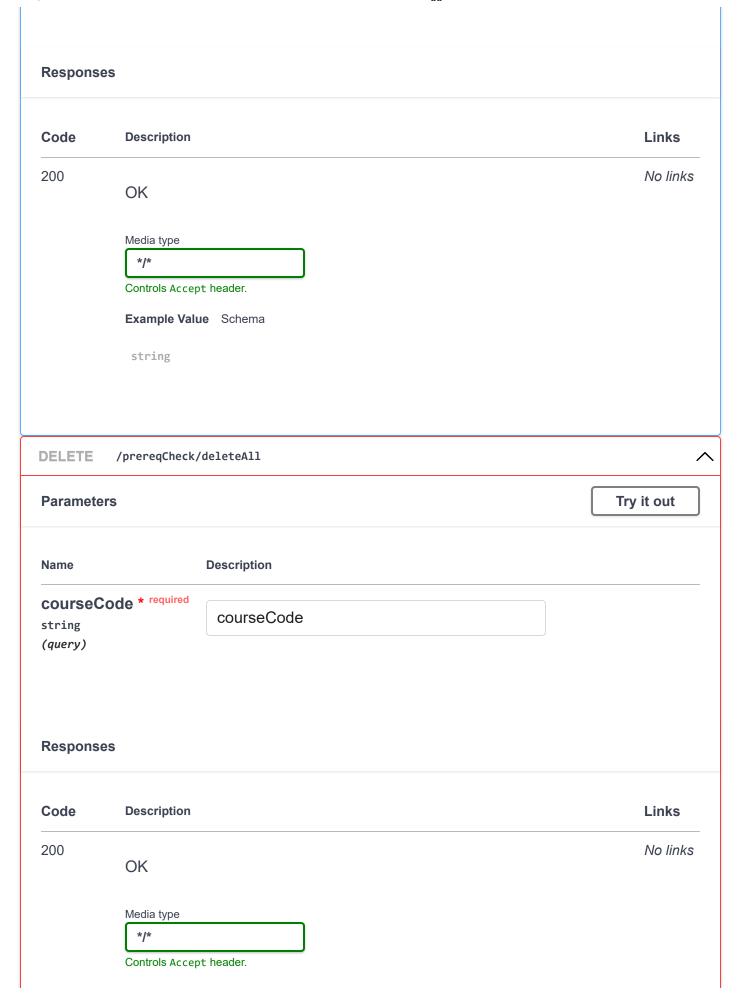


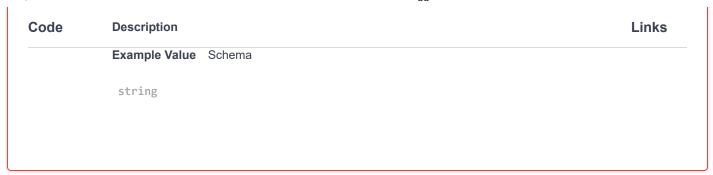


prereqCheck controller controller to experiment with manipulating prerequisites for courses

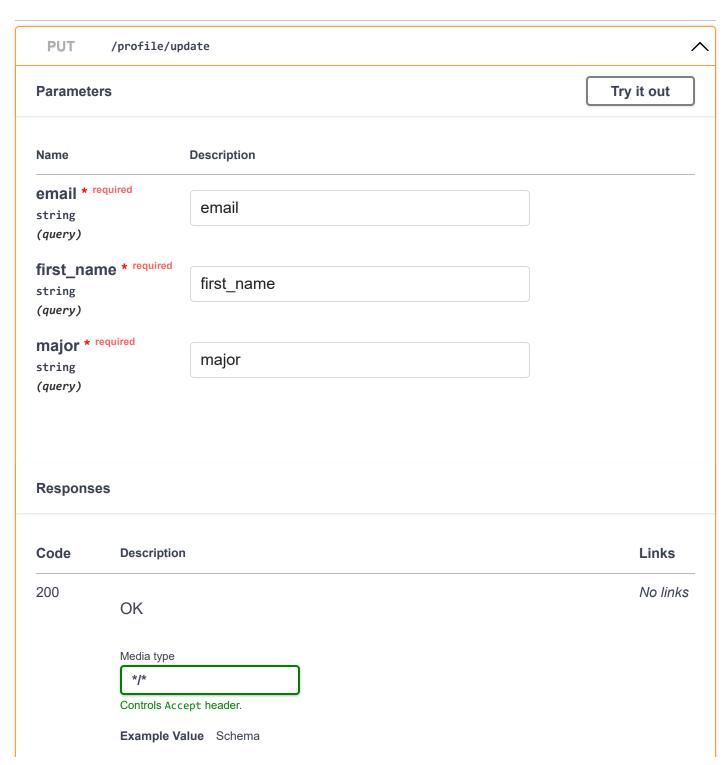
PUT /prereqCheck/add

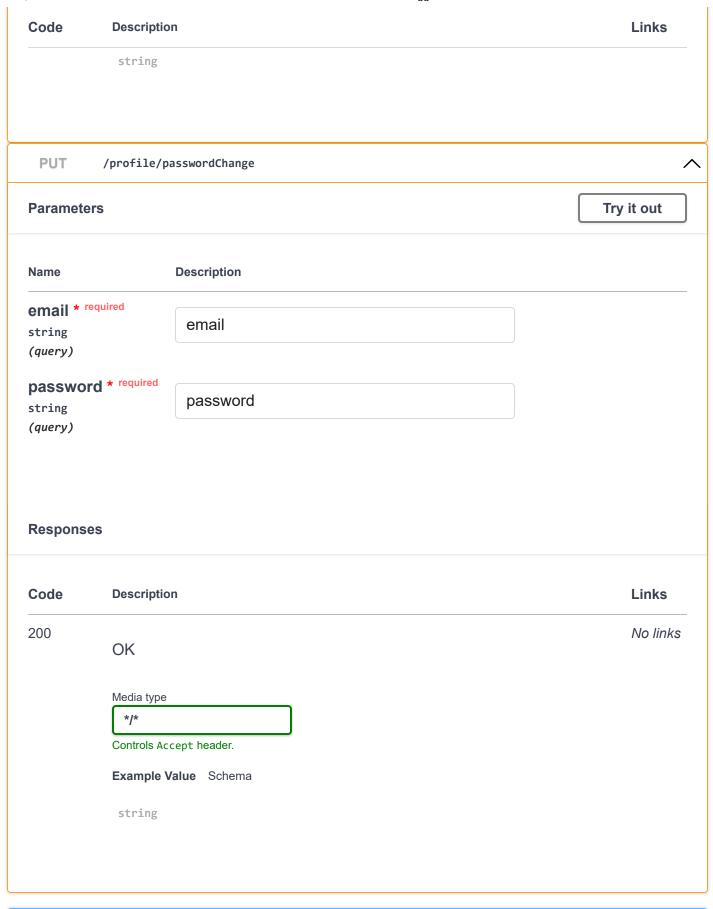






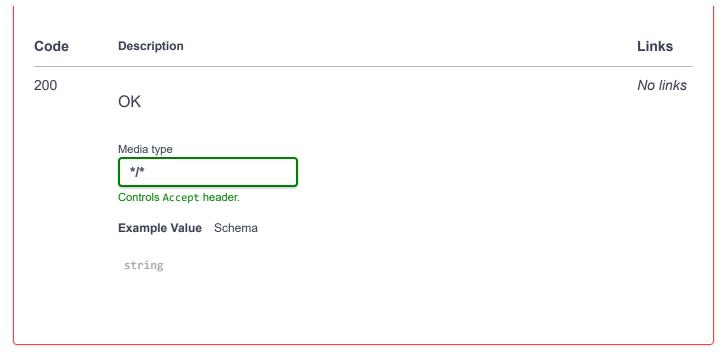
Profile Update Controller controller to deal with all the endpoints relating to updating and displaying user profile information



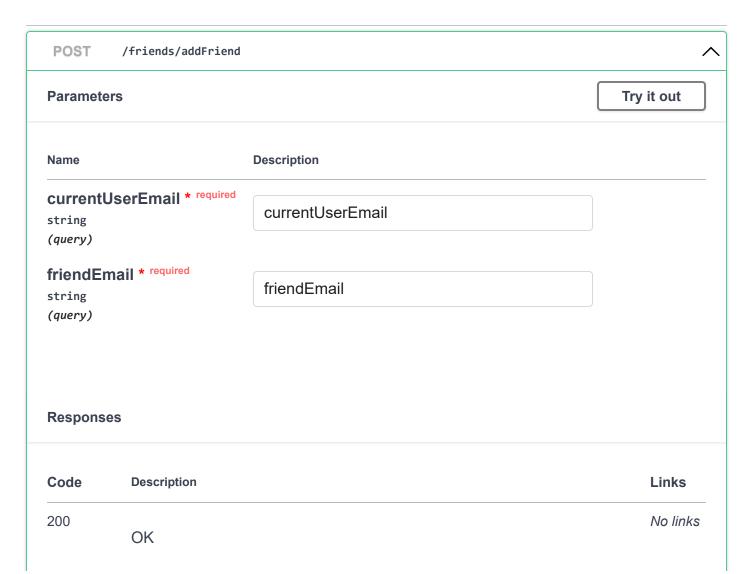


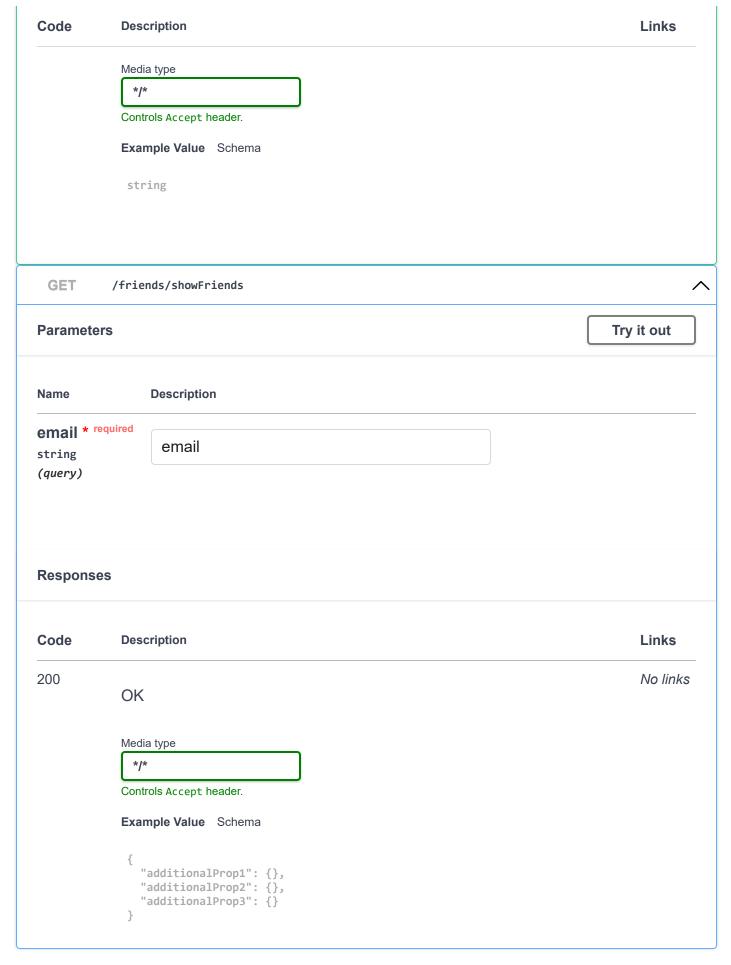
GET /profile/getCred ^

Try it out **Description** Name email * required email string (query) Responses Code **Description** Links 200 No links OK Media type */* Controls Accept header. Example Value Schema "additionalProp1": {}, "additionalProp2": {}, "additionalProp3": {} } **DELETE** /profile/delete Try it out **Parameters** Name **Description** email * required email string (query) Responses



class for all endpoints relating the friends feature of cyplan Friends Controller including updating database & displaying information about





Schemas	^
schedule	
JSONObject	