

MenuWorld API

Method	Endpoint	Description	Input JSON	Output JSON
POST*	/users	Create user	email, userName, password, owner (boolean)	userID, token
GET*	/users/:uid	Get user's name, status, and likes	Nothing	userName, owner (boolean), foods (array of foodID)
GET	/users/:uid/private	Get user's email, recently-viewed meals, and menus (owners only)	{ } (empty JSON)	email, foods (array of foodID), menus (array of menuID)
DELETE	/users/:uid	Suspend user (can be called by the user himself/herself or admin)	{ } (empty JSON)	Just an HTTP OK
PUT		Change password and/or email	oldPassword, [password], [email] (square brackets mean "optional")	Just an HTTP OK
POST	/login	Log in	email, password	userID, token
POST	/menus	Create menu	restaurantName, menuName, address	menuID
GET*	/menus/:mid	Get menu	Nothing	restaurantName, menuName, address, foods (array of foodID)
DELETE		Delete menu	{ } (empty JSON)	Just an HTTP OK
PUT		Modify menu	[restaurantName], [menuName], [address]	Just an HTTP OK
POST	/foods	Add food to menu	menuID, foodName, [photo (in base 64 encoding)], mealType (string), cuisine (string), allergens (array of strings)	foodID

Method	Endpoint	Description	Input JSON	Output JSON
GET*	/foods/:fid[/view] (if "/view", the food will be registered in the user's view history)	Get food details	Nothing	foodName, menuID, photo (in base 64 encoding), likes (array of userID strings) mealType (string), cuisine (string), allergens (array of AR)
PUT	/foods/:fid	Modify food	[foodName], [photo (in base 64 encoding)], [mealType (string)], [cuisine (string)], [allergens (array of AR)]	Just an HTTP OK
DELETE		Delete food	{ } (empty JSON)	
POST		Report (confirm/deny) allergen	allergen (string), confirm (boolean)	
DELETE		Un-report allergen	allergen (string)	
POST	/foods/:fid/like	Like food	{ } (empty JSON)	
DELETE		Un-like food	{ } (empty JSON)	
GET*	/top	Top searches	Nothing	searches (array of strings)
	/search/:search+term/:allergy1+allergy2+allergy3+ ...	Search for foods and menus	Nothing	foods (array of foodID), menus (array of menuID)
	/popular	Popular (i.e. most liked) foods	Nothing	foods (array of foodID)
	/latest	Most recently-posted foods	Nothing	

* Token not required. Other requests: pass token via the **bearer authorization** HTTP header.

AR: allergen (string), confirms (array of userID strings), denys (array of userID strings)

"Allergen" means that the food does not contain the ingredient listed.