Lab Exercise #6: Appendix - REST endpoint documentation

Endpoint[1]	GET[2]	POST[3]	PUT[4]	DELETE[5]
/api/health	Health check (200)			
/api/users		Register user (201, 400, 409)		
/api/auth	Get current session user (200)	Login (200, 401)		Logout (200)
/api/budgets	List budgets for current user (200, 401)	Create budget (201, 400, 401)		
/api/budgets/:id	Get budget details for current user (200, 400, 401, 404)			
/api/budgets/:id/users	List budget members (200, 400, 401, 403)	Invite user to budget (201, 400, 401, 403, 404, 409)		
/api/invites/pending	List pending invites for current user (200, 401)			
/api/invites/sent	List invites sent by current user (200, 401)			

[1] Endpoint of the resource and/or parameters
[2] GET method description and returned HTTP status code with the description.
[3] POST method description and returned HTTP status code with the description.
[4] PUT method description and returned HTTP status code with the description.
[5] DELETE method description and returned HTTP status code with the description.

/api/invites/:inviteId/respond		Respond to invite: ACCEPT/DENY (200, 400, 401, 403, 404, 409)		
/api/notifications	List notifications (supports ?unreadOnly=1) (200, 401)			
/api/notifications/:id/read			PATCH: Mark notification as read (200, 400, 401, 404)	
/api/notifications/read-all			PATCH: Mark all notifications as read (200, 401)	
/api/budgets/:budgetId/transactions	List transactions for a budget (200, 400, 401, 403)	Create transaction (201, 400, 401, 403)		
/api/budgets/:budgetId/transactions/:transactionId			Update transaction (200, 400, 401, 403, 404)	Delete transaction (200, 400, 401, 403, 404)
/api/categories	List categories (200, 401)			
/api/admin/users	Admin: list all users (200, 401, 403)	Admin: create user (201, 400, 401, 403, 409)		
/api/admin/users/:id	Admin: get user by id (200, 400, 401, 403, 404)		Admin: update user (200, 400, 401, 403, 404, 409)	Admin: delete user (200, 400, 401, 403, 404)
/api/admin/roles	Admin: list roles (200, 403)			

/api/admin/categories	Admin: list categories (200, 401, 403)	Admin: create category (201, 400, 401, 403, 409)		
/api/admin/categories/:id			Admin: update category (200, 400, 401, 403, 404, 409)	Admin: delete category (200, 400, 401, 403, 404, 409)
/api/control/budgets	Control: list all budgets (200, 401)			
/api/control/transactions	Control: list all transactions (200, 401)			