What	Front-End	Back-End	Mapping Type	Endpoint URL	Path Variable	Request Body	API returns: (if successful)	Controller Method
Postman Setup			POST	/users/bulk		usersToAdd		addUsers
Postman Setup			POST	/posts		postsToAdd		addPosts
Postman Setup			POST	/tags/bulk		tagsToAdd		addTags
User	Create an account	Add a user to DB	POST	/users		userToAdd	newUserCreated	addUser
User	Display users in profile directory	Fetch all registered users from database	GET	/users			registeredUsersList	getAllUsers
User	Display user info on My Account page	Fetch specific user info	GET	/users/{userId}	{userId}		registeredUsers	getUserById
User	User updates account info	Update user by id on DB	PUT	/users/{userId}	{userId}	editedUser	updatedUser	updateUserById
User	User delete his account	Delete user by id on DB	DELETE	/users/{userId}	{userId}		`success message`	deleteUserById
Address	Display and have user confirm mailing/billing address for merch	Get address for specific user	GET	address/users/{userId}		newAddress	addressForSpecifiedUser	getAddressByUserId
Address	Update just the billing/mailing address for a user	Patch address for that specific user	PATCH	address/users/{userId}	{userId}		modifiedAddressForUserId	modifyAddressForUserId
Tag	User creates a tag	Add that tag to our DB	POST	/tags		tagToAdd	newTagCreated	addTag
Tag	User adds a tag to post (can only add existing tag- and if they have a new one, they must first creat the tag and then add it)	Add a tag to post in DB	POST	/tags/{tagId}/posts/{postId}	{postId} {tagId}		modifiedAddressForUserId	addTagToPost
Tag	To display tags to users to choose from	Fetch tags from DB	GET	/tags			listOfTagsOptions	getAllTags

Tag	Get posts associated with a tag	Fetch posts with that tag from DB	GET	/tags/{tagId}/posts	{tagId}		postsWithTagId	getPostsWithTagId
Tag	User wants to update a tag he/she/they created	Update that tag on DB too to reflect	PUT	/tags/{tagId}	{tagId}		updatedTag	tagToUpdate
Tag	User wants to delete a tag from post		DELETE	tags/{tagId}	{tagId}		`success message`	deleteTagById
Post	User makes a new post	Create new post in DB	POST	/posts/users/{userId}	{userId}	postToAdd	newPostCreated	addPost
Post	To display posts on home page	Fetch posts from DB	GET	/posts			allPosts	getAllPosts
Post	To display posts that match keyword	Fetch posts from DB where desription contains keyword	GET	/posts?keyword={keyword)		* @RequestParam String keyword		getPostsByDescriptionContaining
Post	User edits existing post	Update existing post by Id in DB	PUT	/posts/{postId}	{postId}	editedPost	updatedPost	updatePostById
Delete	User delete his post	Delete a post by post id	DELETE	/posts/{postId}	{postId}		`success message`	deletePostById
Comment	User comments	Add comment to post in DB	POST	comments/posts/{postId}/	{postId}	commentToAdd	newComment	addCommentToAPost
Comment	To display comments tied a post	Fetch comments tied to a post	GET	comments/posts/{postId}/	{postId}		commentsTiedToPost	getCommentsByPost
Comment	User edits comments on a post	Edit that comment for post on backend	PUT	comments/{commentId}/posts/{postId}/	{postId} {commentId}	editedComment	updatedCommentOnPost	updateCommentForPost
Comment	User deletes comment on a post	Remove that comment from that post in DB	DELETE	/comments/{commentId}	{commentId}		`success message`	deleteCommentById