1. Designing the Follows RESTful Web Service API

Use Case	**Method /resource/path**	**Request**	**Response body**

User follows another user	POST
	/users/:uid/follows/:uid2
		User 1 and User 2 PK	{
			User: uid2,
			followedBy: uid1
			id:
			}

User unfollows another user	DELETE
	/users/:uid/follows/:uid2
		User 1 and User 2 PK	{
			"deletedCount":
			}

User views a list of other users they are following	GET /users/:uid/follows/:uid2	User 1 and User 2 PK	Json list of users they are following
User views a list of other users that are following them	GET /users/:uid/following/:uid2	User 1 and User 2 PK	Json list of users that are following them

User unfollows all followers **(NEW)**	DELETE
	/users/:uid/followers
		User PK	{
			"deletedCount":
			}

User is unfollowed by all followers **(NEW)**	DELETE
	/users/:uid/following
		User PK	{
			"deletedCount":
			}

2. Design the Bookmarks RESTful Web Service API

Use Case	**Method /resource/path**	**Request**	**Response body**

User bookmarks a tuit	POST
	/users/:uid/bookmarks/:tuit
		User and Tuit PK	{
			"tuit": tuit,
			"bookmarkedBy": uid,
			"_id":
			}

User unbookmarks a tuit	DELETE
	/users/:uid/bookmarks/:tuit
		User and Tuit PK	{
			"deletedCount":
			}

User views a list of tuits they have bookmarked	GET /users/:uid/bookmarks/:tuit	User and Tuit PK	Json list of

User unbookmarks all tuits **(NEW)**	DELETE /users/:uid/bookmarks	User PK	{
			"deletedCount":
			}

To find all users that have bookmarked a certain tuit **(NEW)**	GET tuits/uid/bookmarks	User PK	Json list of users bookmarked a tuit

3. Design the Messages RESTful Web Service API

Use Case	**Method /resource/path**	**Request**	**Response body**

User sends a message to another user	POST /users/:uid/messages/:uid2	User 1 and User 2 PK	{
			Message: “ “,
			To : uid,
			From: uid2,
			sentOn: “ “ ,
			Mid: “ “
			}

User views a list of messages they have sent	GET /users/:uid/sentmessages/:uid2	User 1 and User 2 PK	Json list of messages sent by user
User views a list of messages sent to them	GET /users/:uid/messages/:uid2	User 1 and User 2 PK	Json list of messages received by user

User deletes a message	DELETE /users/:uid/messages/:uid2	User 1 and User 2 PK	{
			"deletedCount":
			}

User can find a message in chatbox **(NEW)**	GET /messages/:mid	Message-id	Message array

User deletes all sent messages **(NEW)**	DELETE /users/uid/sentmessages	User id	{
			"deletedCount":
			}