

## Server Side API

所有API返回的格式是JSON，最好能把status和error加到response header里。这样返回的JSON里就不用包括了。

带下划线的是custom data structure，具体结构在下面，加粗的是数组

1. register  
method: POST  
argument: email, encrypt\_password, first\_name, last\_name, nickname, avatar\_data  
return: status, [error], user
2. loginWithFB  
method: POST  
argument: fb\_token  
return: status, [error], user, token
3. loginWithTwitter  
method: POST  
argument: tw\_token  
return: status, [error], user, token
4. loginWithPassword  
method: POST  
argument: email, encrypt\_password  
return: status, [error], user, token  
传递过来的password已经是加密后的，可以直接与数据库中的比对不用再做处理
5. bindFB  
method: POST  
argument: fb\_token, token  
return: status, [error]
6. bind3D  
method: POST  
argument: email, encrypt\_password, token  
return: status, [error]  
传递过来的password已经是加密后的，可以直接添加到数据库中
7. bindTwitter  
method: POST  
argument: twitter\_token, token  
return: status, [error]
8. createItem  
method: POST  
argument: token, item\_name  
return: status, [error], item\_name, item\_id
9. addResource  
method: POST  
argument: token, item\_id, resource\_type, resource\_data  
return: status, [error], item\_id, resource
10. getItem  
method: GET  
argument: token, item\_id, [thumbnail=false]  
return: status, [error], item
11. getFollowers  
method: GET  
argument: token  
return: status, [error], **users**
12. getFollowees  
method: GET  
argument: token  
return: status, [error], **users**

13. getFeeds  
method: GET  
argument: token, [index = 0], [batch\_size = 20]  
return: status, [error], **feeds**
14. getFeedDetail  
method: GET  
argument: token, item\_id  
return: status, [error], **feed\_detail**  
feed\_detail的comments仅包括前20条comment, 如果想要拿更多客户端会再调用getComments
15. getComments  
method: GET  
argument: token, item\_id, [index = 0], [batch\_size = 10]  
return: status, [error], **comments**  
startIndex = batch\_size \* index; endIndex = batch\_size \* (index + 1), 包含startIndex的comment但不包含endIndex的Comment
16. createComment  
method: POST  
argument: token, item\_id, content  
return: status, [error], **comment**
17. createLike  
method: POST  
argument: token, item\_id  
return: status, [error], **like**
18. logout  
method: POST  
argument: token  
return: status, [error]

## Data Structures

### user

uid  
nickname  
email  
avatar\_url  
first\_name  
last\_name

### item

item\_id  
user

### resources

### resource

resource\_id  
type  
url

### like

like\_id  
user

### feed

item  
like\_count  
comment\_count  
share\_count

## **comment**

comment\_id  
user  
index  
created\_date

## **feed\_detail**

item  
like\_count  
comment\_count  
share\_count  
comments