CMS Antiques API Documentation

Table of Contents

[Authentication/Account API 2](#_Toc57121097)

[GET /login 2](#_Toc57121098)

[GET /login-error 3](#_Toc57121099)

[GET /member/accountProfile/ 3](#_Toc57121100)

[POST /member/accountProfile/ 3](#_Toc57121101)

[GET /member/resetPassword 4](#_Toc57121102)

[POST /member/resetPassword 4](#_Toc57121103)

[GET /register 5](#_Toc57121104)

[POST /register 5](#_Toc57121105)

[Classifieds API 5](#_Toc57121106)

[POST /add\_tag 5](#_Toc57121107)

[GET /classifieds\_ads (redirect) 7](#_Toc57121108)

[POST /classifieds\_ads (postClassifieds) 7](#_Toc57121109)

[GET /classifieds\_ads/{category}/{page} 8](#_Toc57121110)

[GET /classifieds\_ads/edit/{id} 8](#_Toc57121111)

[GET /classifieds\_ads/new 9](#_Toc57121112)

[GET /classifieds\_ads/view/{id} 9](#_Toc57121113)

[POST /delete\_classified 10](#_Toc57121114)

[Posts API 11](#_Toc57121115)

[GET /posts/{category}/{page} 11](#_Toc57121116)

[GET /posts/newpost 12](#_Toc57121117)

[GET /posts/view/{id} 12](#_Toc57121118)

[POST /posts/view/deletepost\_{id} 12](#_Toc57121119)

[POST /posts/view/editpost\_{id} (create new post, update old post) 13](#_Toc57121120)

Administrator API [1](#_Toc57121115)5

[GET /admin/viewRestricted/getUsersTable 1](#_Toc57121116)5

[GET /admin/viewRestricted/viewUsers 1](#_Toc57121117)5

[GET /admin/viewRestricted/getRolesTable 1](#_Toc57121118)5

[GET /admin/viewRestricted/viewRoles 1](#_Toc57121119)5

[GET /admin/viewRestricted/getPermissionsTable 1](#_Toc57121120)6

[GET /admin/viewRestricted/viewPermissions 1](#_Toc57121120)6

[GET /admin/viewRestricted/getStatesTable 1](#_Toc57121117)6

[GET /admin/viewRestricted/viewStates 1](#_Toc57121118)6

[GET /admin/viewRestricted/getCategoriesTable 1](#_Toc57121119)7

[GET /admin/viewRestricted/viewCategories 1](#_Toc57121120)7

[GET /admin/createUpdate/user 1](#_Toc57121117)7

[POST /admin/createUpdate/user 1](#_Toc57121118)7

[DELETE /admin/createUpdate/user 1](#_Toc57121119)8

[GET /admin/createUpdate/role 1](#_Toc57121117)8

[POST /admin/createUpdate/role 1](#_Toc57121118)9

[DELETE /admin/createUpdate/role……………………………………………………………………………………………………1](#_Toc57121119)9

[GET /admin/createUpdate/permission 1](#_Toc57121117)9

[POST /admin/createUpdate/permission 2](#_Toc57121118)0

[DELETE/admin/createUpdate/permission………………………………………………………………………………………….2](#_Toc57121119)0

[GET /admin/createUpdate/category 2](#_Toc57121117)1

[POST /admin/createUpdate/category 2](#_Toc57121118)1

[DELETE admin/createUpdate/category………………………………………………………………………………………………2](#_Toc57121119)1

[GET /admin/createUpdate/state……………………………………………………………………………………………………….22](#_Toc57121117)

[POST /admin/createUpdate/state……………………………………………………………………………………………………..22](#_Toc57121118)

[DELETE admin/createUpdate/state……………………………………………………………………………………………………22](#_Toc57121119)

# Authentication/Account API

## GET /login

Scopes: All  
Parameters: None  
Authorization: None  
Responses:

200:

## GET /login-error

Scopes: All  
Parameters: None  
Authorization: None  
Responses:

200:

## GET /member/accountProfile/

Scopes: Member:Admin  
Authorization: JWT  
Responses:

200:

## POST /member/accountProfile/

Scopes: Member:Admin  
Parameters:

1. accountNonExpired
   1. boolean
2. accountNonLocked
   1. boolean
3. address.address
   1. string
4. address.city.createdOn
   1. string($date-time)
5. address.city.id
   1. string($uuid)
6. address.city.modifiedOn
   1. string($date-time)
7. address.city.name
   1. string
8. address.city.postalCode
   1. string
9. address.city.state.createdOn
   1. string($date-time)
10. address.city.state.id
    1. string($uuid)
11. address.city.state.modifiedOn
    1. string($date-time)
12. address.city.state.name
    1. string
13. address.createdOn
    1. string($date-time)
14. address.id
    1. string($uuid)
15. address.modifiedOn
    1. string($date-time)
16. authorities[0].authority
    1. string
17. createdOn
    1. string($date-time)
18. credentialsExpiredOn
    1. string($date-time)
19. credentialsNonExpired
    1. boolean
20. email
    1. string
21. enabled
    1. boolean
22. expiredOn
    1. string($date-time)
23. firstName
    1. string
24. id
    1. string($uuid)
25. imagePath
    1. string
26. lastName
    1. string
27. locked
    1. boolean
28. modifiedOn
    1. string($date-time)
29. password
    1. string
30. passwordConfirm
    1. string
31. permissions[0].createdOn
    1. string($date-time)
32. permissions[0].id
    1. string($uuid)
33. permissions[0].modifiedOn
    1. string($date-time)
34. permissions[0].name
    1. string
35. phoneNum
    1. string
36. username
    1. string

Authorization: JWT  
Responses:

200:

Authorization: JWT  
Responses:

200:

## GET /member/resetPassword

Scopes: Member:Admin  
Parameters: None  
Authorization: JWT  
Responses:

200:

## POST /member/resetPassword

Scopes: Member:Admin  
Parameters:

1. currentPasword
   1. string
2. encryptedCurrentPassword
   1. string
3. password
   1. string
4. passwordConfirm
   1. string
5. userName
   1. string

Authorization: JWT  
Responses:

200:

## GET /register

Scopes: Member:Admin  
Parameters: None  
Authorization: JWT  
Responses:

200:

## POST /register

Scopes:Member:Admin

Parameters:

1. username
2. String
3. firstName
4. String
5. lastName
6. String
7. Email
8. String
9. password
10. String
11. passwordConfirm
12. String

Authorization: JWT

Responses:

200:

# Classifieds API

## POST /add\_tag

Scopes: Member:Admin  
Parameters:

1. creator.accountNonExpired
   1. boolean
2. creator.accountNonLocked
   1. boolean
3. creator.address.address
   1. string
4. creator.address.city.createdOn
   1. string($date-time)
5. creator.address.city.id
   1. string($uuid)
6. creator.address.city.modifiedOn
   1. string($date-time)
7. creator.address.city.name
   1. string
8. creator.address.city.postalCode
   1. string
9. creator.address.city.state.createdOn
   1. string($date-time)
10. creator.address.city.state.id
    1. string($uuid)
11. creator.address.city.state.modifiedOn
    1. string($date-time)
12. creator.address.city.state.name
    1. string
13. creator.address.createdOn
    1. string($date-time)
14. creator.address.id
    1. string($uuid)
15. creator.address.modifiedOn
    1. string($date-time)
16. creator.authorities[0].authority
    1. string
17. creator.createdOn
    1. string($date-time)
18. creator.credentialsExpiredOn
    1. string($date-time)
19. creator.credentialsNonExpired
    1. boolean
20. creator.email
    1. string
21. creator.enabled
    1. boolean
22. creator.expiredOn
    1. string($date-time)
23. creator.firstName
    1. string
24. creator.id
    1. string($uuid)
25. creator.imagePath
    1. string
26. creator.lastName
    1. string
27. creator.locked
    1. boolean
28. creator.modifiedOn
    1. string($date-time)
29. creator.password
    1. string
30. creator.passwordConfirm
    1. string
31. creator.permissions[0].createdOn
    1. string($date-time)
32. creator.permissions[0].id
    1. string($uuid)
33. creator.permissions[0].modifiedOn
    1. string($date-time)
34. creator.permissions[0].name
    1. string
35. creator.phoneNum
    1. string
36. creator.username
    1. string
37. description
    1. string
38. endDate
    1. string($date-time)
39. id
    1. string($uuid)
40. item.category
    1. string
41. item.id
    1. string($uuid)
42. item.itemImage.fileName
    1. string
43. item.itemImage.filePath
    1. string
44. item.itemImage.id
    1. string($uuid)
45. item.name
    1. string
46. price
    1. string
47. tag
    1. application/json
48. tags
    1. array[string]
49. title
    1. string

Authorization: JWT  
Responses:

200:

## GET /classifieds\_ads (redirect)

Scopes: Member  
Parameters: None  
Authorization: JWT  
Responses:

200:

## POST /classifieds\_ads (postClassifieds)

Scopes: Member  
Parameters:

1. creator.accountNonExpired
   1. boolean
2. creator.accountNonLocked
   1. boolean
3. creator.address.address
   1. string
4. creator.address.city.createdOn
   1. string($date-time)
5. creator.address.city.id
   1. string($uuid)
6. creator.address.city.modifiedOn
   1. string($date-time)
7. creator.address.city.name
   1. string
8. creator.address.city.postalCode
   1. string
9. creator.address.city.state.createdOn
   1. string($date-time)
10. creator.address.city.state.id
    1. string($uuid)
11. creator.address.city.state.modifiedOn
    1. string($date-time)
12. creator.address.city.state.name
    1. string
13. creator.address.createdOn
    1. string($date-time)
14. creator.address.id
    1. string($uuid)
15. creator.address.modifiedOn
    1. string($date-time)
16. creator.authorities[0].authority
    1. string
17. creator.createdOn
    1. string($date-time)
18. creator.credentialsExpiredOn
    1. string($date-time)
19. creator.credentialsNonExpired
    1. boolean
20. creator.email
    1. string
21. creator.enabled
    1. boolean
22. creator.expiredOn
    1. string($date-time)
23. creator.firstName
    1. string
24. creator.id
    1. string($uuid)
25. creator.imagePath
    1. string
26. creator.lastName
    1. string
27. creator.locked
    1. boolean
28. creator.modifiedOn
    1. string($date-time)
29. creator.password
    1. string
30. creator.passwordConfirm
    1. string
31. creator.permissions[0].createdOn
    1. string($date-time)
32. creator.permissions[0].id
    1. string($uuid)
33. creator.permissions[0].modifiedOn
    1. string($date-time)
34. creator.permissions[0].name
    1. string
35. creator.phoneNum
    1. string
36. creator.username
    1. string
37. description
    1. string
38. endDate
    1. string($date-time)
39. id
    1. string($uuid)
40. item.category
    1. string
41. item.id
    1. string($uuid)
42. item.itemImage.fileName
    1. string
43. item.itemImage.filePath
    1. string
44. item.itemImage.id
    1. string($uuid)
45. item.name
    1. string
46. price
    1. string
47. tags
    1. array[string]
48. title
    1. string

Authorization: JWT  
Responses:

200:

## GET /classifieds\_ads/{category}/{page}

Scopes: Member  
Parameters:

1. category
   1. string
2. page
   1. string
3. search
   1. string

Authorization: JWT  
Responses:

200:

## GET /classifieds\_ads/edit/{id}

Scopes: Member:Admin  
Parameters:

1. authenticated
   1. boolean
2. authorities[0].authority
   1. string
3. credentials
   1. object
4. details
   1. object
5. id \*
   1. string
6. principal
   1. object

Authorization: JWT  
Responses:

200:

## GET /classifieds\_ads/new

Scopes: Member:Admin  
Parameters:

1. authenticated
   1. boolean
2. authorities[0].authority
   1. string
3. credentials
   1. object
4. details
   1. object
5. id \*
   1. string
6. principal
   1. object

Authorization: JWT  
Responses:

200:

## GET /classifieds\_ads/view/{id}

Scopes: Member:Admin  
Parameters:

1. authenticated
   1. boolean
2. authorities[0].authority
   1. string
3. credentials
   1. object
4. details
   1. object
5. id \*
   1. string
6. principal
   1. object

Authorization: JWT  
Responses:

200:

## POST /delete\_classified

Scopes: Member:Admin  
Parameters:

1. creator.accountNonExpired
   1. boolean
2. creator.accountNonLocked
   1. boolean
3. creator.address.address
   1. string
4. creator.address.city.createdOn
   1. string($date-time)
5. creator.address.city.id
   1. string($uuid)
6. creator.address.city.modifiedOn
   1. string($date-time)
7. creator.address.city.name
   1. string
8. creator.address.city.postalCode
   1. string
9. creator.address.city.state.createdOn
   1. string($date-time)
10. creator.address.city.state.id
    1. string($uuid)
11. creator.address.city.state.modifiedOn
    1. string($date-time)
12. creator.address.city.state.name
    1. string
13. creator.address.createdOn
    1. string($date-time)
14. creator.address.id
    1. string($uuid)
15. creator.address.modifiedOn
    1. string($date-time)
16. creator.authorities[0].authority
    1. string
17. creator.createdOn
    1. string($date-time)
18. creator.credentialsExpiredOn
    1. string($date-time)
19. creator.credentialsNonExpired
    1. boolean
20. creator.email
    1. string
21. creator.enabled
    1. boolean
22. creator.expiredOn
    1. string($date-time)
23. creator.firstName
    1. string
24. creator.id
    1. string($uuid)
25. creator.imagePath
    1. string
26. creator.lastName
    1. string
27. creator.locked
    1. boolean
28. creator.modifiedOn
    1. string($date-time)
29. creator.password
    1. string
30. creator.passwordConfirm
    1. string
31. creator.permissions[0].createdOn
    1. string($date-time)
32. creator.permissions[0].id
    1. string($uuid)
33. creator.permissions[0].modifiedOn
    1. string($date-time)
34. creator.permissions[0].name
    1. string
35. creator.phoneNum
    1. string
36. creator.username
    1. string
37. description
    1. string
38. endDate
    1. string($date-time)
39. id
    1. string($uuid)
40. item.category
    1. string
41. item.id
    1. string($uuid)
42. item.itemImage.fileName
    1. string
43. item.itemImage.filePath
    1. string
44. item.itemImage.id
    1. string($uuid)
45. item.name
    1. string
46. price
    1. string
47. tags
    1. array[string]
48. title
    1. string

Authorization: JWT  
Responses:

200:

# Posts API

## GET /posts/{category}/{page}

Scopes: All  
Parameters:

1. category \*
   1. string
2. category
   1. page \*
3. string
   1. page
4. searchIN
   1. string

Authorization: None  
Responses:

200:

## GET /posts/newpost

Scopes: Member  
Parameters: id (String)  
Authorization: JWT  
Responses:

200:

## GET /posts/view/{id}

Scopes: All  
Parameters:

1. id
   1. String

Authorization: JWT  
Responses:

200:

## POST /posts/view/deletepost\_{id}

Scopes: Member:Admin  
Parameters:

1. creator.accountNonExpired
   1. boolean
2. creator.accountNonLocked
   1. boolean
3. creator.address.address
   1. string
4. creator.address.city.createdOn
   1. string($date-time)
5. creator.address.city.id
   1. string($uuid)
6. creator.address.city.modifiedOn
   1. string($date-time)
7. creator.address.city.name
   1. string
8. creator.address.city.postalCode
   1. string
9. creator.address.city.state.createdOn
   1. string($date-time)
10. creator.address.city.state.id
    1. string($uuid)
11. creator.address.city.state.modifiedOn
    1. string($date-time)
12. creator.address.city.state.name
    1. string
13. creator.address.createdOn
    1. string($date-time)
14. creator.address.id
    1. string($uuid)
15. creator.address.modifiedOn
    1. string($date-time)
16. creator.authorities[0].authority
    1. string
17. creator.createdOn
    1. string($date-time)
18. creator.credentialsExpiredOn
    1. string($date-time)
19. creator.credentialsNonExpired
    1. boolean
20. creator.email
    1. string
21. creator.enabled
    1. boolean
22. creator.expiredOn
    1. string($date-time)
23. creator.firstName
    1. string
24. creator.id
    1. string($uuid)
25. creator.imagePath
    1. string
26. creator.lastName
    1. string
27. creator.locked
    1. boolean
28. creator.modifiedOn
    1. string($date-time)
29. creator.password
    1. string
30. creator.passwordConfirm
    1. string
31. creator.permissions[0].createdOn
    1. string($date-time)
32. creator.permissions[0].id
    1. string($uuid)
33. creator.permissions[0].modifiedOn
    1. string($date-time)
34. creator.permissions[0].name
    1. string
35. creator.phoneNum
    1. string
36. creator.username
    1. string
37. description
    1. string
38. endDate
    1. string($date-time)
39. id
    1. string($uuid)
40. item.category
    1. string
41. item.id
    1. string($uuid)
42. item.itemImage.fileName
    1. string
43. item.itemImage.filePath
    1. string
44. item.itemImage.id
    1. string($uuid)
45. item.name
    1. string
46. price
    1. string
47. tags
    1. array[string]
48. title
49. a. string

Authorization: JWT  
Responses:

200:

## POST /posts/view/editpost\_{id} (create new post, update old post)

Scopes: Member:Admin  
Parameters:

1. creator.accountNonExpired
   1. boolean
2. creator.accountNonLocked
   1. boolean
3. creator.address.address
   1. string
4. creator.address.city.createdOn
   1. string($date-time)
5. creator.address.city.id
   1. string($uuid)
6. creator.address.city.modifiedOn
   1. string($date-time)
7. creator.address.city.name
   1. string
8. creator.address.city.postalCode
   1. string
9. creator.address.city.state.createdOn
   1. string($date-time)
10. creator.address.city.state.id
    1. string($uuid)
11. creator.address.city.state.modifiedOn
    1. string($date-time)
12. creator.address.city.state.name
    1. string
13. creator.address.createdOn
    1. string($date-time)
14. creator.address.id
    1. string($uuid)
15. creator.address.modifiedOn
    1. string($date-time)
16. creator.authorities[0].authority
    1. string
17. creator.createdOn
    1. string($date-time)
18. creator.credentialsExpiredOn
    1. string($date-time)
19. creator.credentialsNonExpired
    1. boolean
20. creator.email
    1. string
21. creator.enabled
    1. boolean
22. creator.expiredOn
    1. string($date-time)
23. creator.firstName
    1. string
24. creator.id
    1. string($uuid)
25. creator.imagePath
    1. string
26. creator.lastName
    1. string
27. creator.locked
    1. boolean
28. creator.modifiedOn
    1. string($date-time)
29. creator.password
    1. string
30. creator.passwordConfirm
    1. string
31. creator.permissions[0].createdOn
    1. string($date-time)
32. creator.permissions[0].id
    1. string($uuid)
33. creator.permissions[0].modifiedOn
    1. string($date-time)
34. creator.permissions[0].name
    1. string
35. creator.phoneNum
    1. string
36. creator.username
    1. string
37. description
    1. string
38. endDate
    1. string($date-time)
39. id
    1. string($uuid)
40. item.category
    1. string
41. item.id
    1. string($uuid)
42. item.itemImage.fileName
    1. string
43. item.itemImage.filePath
    1. string
44. item.itemImage.id
    1. string($uuid)
45. item.name
    1. string
46. price
    1. string
47. tags
    1. array[string]
48. title
49. 2. a. string
50. Authorization: JWT  
    Responses:

200:

# Administrator API

## GET /admin/viewRestricted/getUsersTable

Scopes: Permission – View\_Restricted

Parameter: none

Authorization: JWT  
Responses: Returns a list of all User objects

200:

## GET /admin/viewRestricted/viewUsers

Scopes: Permission – View\_Restricted

Parameter: none

Authorization: JWT  
Responses:

200:

## GET /admin/viewRestricted/getRolesTable

Scopes: Permission – View\_Restricted

Parameter: none

Authorization: JWT  
Responses:

200: Returns a list of all Role objects

## GET /admin/viewRestricted/viewRoles

Scopes: Permission – View\_Restricted

Parameter: none

Authorization: JWT  
Responses:

200:

## GET /admin/viewRestricted/getPermissionsTable

Scopes: Permission – View\_Restricted

Parameter: none

Authorization: JWT  
Responses:

200: Returns list of all Permission objects

## GET /admin/viewRestricted/viewPermissions

Scopes: Permission – View\_Restricted

Parameter: none

Authorization: JWT  
Responses:

200:

## GET /admin/viewRestricted/getStatesTable

Scopes: Permission – View\_Restricted

Parameter: none

Authorization: JWT  
Responses:

200: Returns list of all State objects

## GET /admin/viewRestricted/viewStates

Scopes: Permission – View\_Restricted

Parameter: none

Authorization: JWT  
Responses:

200:

## GET /admin/viewRestricted/getCategoriesTable

Scopes: Permission – View\_Restricted

Parameter: none

Authorization: JWT  
Responses:

200: Returns list of all Category objects

## GET /admin/viewRestricted/viewCategories

Scopes: Permission – View\_Restricted

Parameters: none

Authorization: JWT  
Responses:

200:

## GET /admin/createUpdate/user/{id}

Scopes:Permission – Modify\_User

Parameters:

1. id
   1. String($uuid)

Authorization: JWT

Responses:

200: Returns a User object by id

## POST /admin/createUpdate/user/{id}(redirect)

Scopes:Permission – Modify\_User

Parameters:

1. Id
2. String($uuid)
3. fName
4. String
5. lName
6. String
7. username
8. String
9. email
10. String
11. password
12. String
13. passwordConfirm
14. String
15. previousUsername
16. String
17. previousEmail
18. String
19. User.role[].id
    1. String($uuid)
20. User.role[].name
    1. String
21. isLocked
    1. boolean

Authorization: JWT

Responses:

200

DELETE /admin/createUpdate/user/{id}

Scopes: Permission – Modify\_User

Parameters:

1. id
2. String($uuid)

Authorization: JWT

Responses:

200: returns a boolean true if successful delete false otherwise

## GET /admin/createUpdate/role/{id}

Scopes:Permission – Modify\_Role

Parameters:

1. id
2. String($uuid)

Authorization: JWT

Responses:

200: Returns a Role object by id

## POST /admin/createUpdate/role/{id}(redirect)

Scopes:Permission – Modify\_Role

Parameters:

1. id
2. String($uuid)
3. name
4. String($uuid)
5. createdOn
   1. String($date-time)
6. role.permission[].id
   1. String($uuid)
7. Role.permission[].name
   1. String

Authorization: JWT

Responses:

200

DELETE /admin/createUpdate/role/{id}

Scopes: Permission – Modify\_Role

Parameters:

1. id
2. String($uuid)

Authorization: JWT

Responses:

200: returns a boolean true if successful delete false otherwise

## GET /admin/createUpdate/permission/{id}

Scopes:Permission – Modify\_Permission

Parameters:

1. id
2. String($uuid)

Authorization: JWT

Responses:

200: Returns a Permission object by id

## POST /admin/createUpdate/permission/{id}(redirect)

Scopes:Permission – Modify\_Permission

Parameters:

1. id
2. String($uuid)
3. name
4. String($uuid)
5. createdOn
   1. String($date-time)
6. permission.role[].id
   1. String($uuid)
7. permission.role[].name
   1. String

Authorization: JWT

Responses:

200

DELETE /admin/createUpdate/permission/{id}

Scopes: Permission – Modify\_Permission

Parameters:

1. id
2. String($uuid)

Authorization: JWT

Responses:

200: returns a boolean true if successful delete false otherwise

## GET /admin/createUpdate/category/{id}

Scopes:Permission – Modify\_Category

Parameters:

1. id
2. String($uuid)

Authorization: JWT

Responses:

200: Returns a Category object by id

## POST /admin/createUpdate/category/{id}(redirect)

Scopes:Permission – Modify\_Category

Parameters:

1. id
2. String($uuid)
3. name
4. String($uuid)
5. createdOn
   1. String($date-time)

Authorization: JWT

Responses:

200

DELETE /admin/createUpdate/category/{id}

Scopes: Permission – Modify\_Category

Parameters:

1. id
2. String($uuid)

Authorization: JWT

Responses:

200: returns a boolean true if successful delete false otherwise

## GET /admin/createUpdate/state/{id}

Scopes:Permission – Modify\_State

Parameters:

1. id
2. String($uuid)

Authorization: JWT

Responses:

200: Returns a State object by id

## POST /admin/createUpdate/state/{id}(redirect)

Scopes:Permission – Modify\_State

Parameters:

1. id
2. String($uuid)
3. name
4. String($uuid)
5. createdOn
   1. String($date-time)

Authorization: JWT

Responses:

200

DELETE /admin/createUpdate/state/{id}

Scopes: Permission – Modify\_State

Parameters:

1. id
2. String($uuid)

Authorization: JWT

Responses:

200: returns a boolean true if successful delete false otherwise