Unit Testing

Class UserMemberControllerTest

Unit Test Case (UTC01): testGetMember() : List<Member>

Description

Test testGetMember method by getting list of member information in the database.

Test Function

Public List<Member>getMember()

Test Data

\*member1 and member2 are in database.

|  |  |  |  |
| --- | --- | --- | --- |
| **Object Name** | **Id** | **Email** | **Password** |
| Member1 | 1 | "se542115004@vr.camt.info" | "1234" |
| Memver2 | 2 | "se552115065@vr.camt.onfo" | "5544" |

Test Cases

|  |  |  |
| --- | --- | --- |
| **Case** | **Description** | **Expected Result** |
| **Type: List of Member** |
| 1 | Test getting all of Member. | { member1 and member2 } |

Unit Test Case (UTC02): testGetMemberByEmail() : Member List<Member>

Description

Test testGetMemberByEmail method by getting member information in the database by using member email.

Test Function

Public getMemberByEmail(email : string)

Test Data

\*member1 and member2 are in database.

|  |  |  |
| --- | --- | --- |
| **Object Name** | **Id** | **Email** |
| Member1 | 1 | "se542115004@vr.camt.info" |
| Memver2 | 2 | "se552115065@vr.camt.onfo" |

Test Cases

|  |  |  |
| --- | --- | --- |
| **Case** | **Description** | **Expected Result** |
| **Type: String Email** |
| 1 | Test getting Member by inputting correct email. | { email1,email2 } |

Unit Test Case (UTC03): testAddMemberByEmail() : Member

Description

Test testAddMemberByEmail method by inputting member information in the database by using member email.

Test Function

Public Member addMember(member : Member)

Test Data

|  |  |  |  |
| --- | --- | --- | --- |
| **Object Name** | **Id** | **Name** | **Email** |
| Member1 | 1 | "Krittika Yothawuth" | "se542115004@vr.camt.info" |

Test Cases

|  |  |  |  |
| --- | --- | --- | --- |
| **Case** | **Description** | **Input** | **Expected Results** |
| **Type: User object** | **Type: member** |
| 1 | Test with all parameters have input the data | Member1 | True |
| 2 | Test with input duplicate data | Member1 | False |
| 3 | Test with empty data | new member1() | False |
| 4 | Test with null | Null | False |

Unit Test Case (UTC04): testLogin() : Member

Description

Test testLogin method by getting list of user information in the database by using member email and member password.

Test Function

Public Member login(email : string, password : string)

Test Data

\*member1 in database.

|  |  |  |  |
| --- | --- | --- | --- |
| **Object Name** | **Id** | **Email** | **Password** |
| Member1 | 1 | "se542115004@vr.camt.info" | "1234" |

Test Cases

|  |  |  |  |
| --- | --- | --- | --- |
| **Case** | **Description** | **Input** | **Expected Results** |
| **Type: User object** | **Type: member** |
| 1 | Test with all parameters have input the data | Member1 | True |
| 2 | Test with input duplicate data | Member1 | False |
| 3 | Test with empty data | new member1() | False |
| 4 | Test with null | Null | False |

Unit Test Case (UTC05): testGetAdmin() : List<Admin>

Description

Test testGetMember method by getting list of admin information in the database.

Test Function

Public List<Admin>getAdmin()

Test Data

\*admin1 and admin2 are in database.

|  |  |  |
| --- | --- | --- |
| **Object Name** | **Id** | **Email** |
| Admin1 | 1 | "se542115004@vr.camt.info" |
| Admin2 | 2 | "se552115065@vr.camt.onfo" |

Test Cases

|  |  |  |
| --- | --- | --- |
| **Case** | **Description** | **Expected Result** |
| **Type: List of Admin** |
| 1 | Test getting all of Admin. | { admin1 } |

Unit Test Case (UTC06): testGetAdminByEmail() : Admin

Description

Test testGetAdminByEmail method by getting admin information in the database by using admin email.

Test Function

Public getAdminByEmail(email : string)

Test Data

\*admin1 and admin2 are in database.

|  |  |  |
| --- | --- | --- |
| **Object Name** | **Id** | **Email** |
| Admin1 | 1 | "se542115004@vr.camt.info" |
| Admin2 | 2 | "se552115065@vr.camt.onfo" |

Test Cases

|  |  |  |
| --- | --- | --- |
| **Case** | **Description** | **Expected Result** |
| **Type: String Email** |
| 1 | Test getting Member by inputting correct email. | { email1.eamil2 } |

Unit Test Case (UTC07): testAddAdminByEmail() : Admin

Description

Test testAddAdminByEmail method by inputting member information in the database by using admin email.

Test Function

Public Member addAdmin(Admin : Admin)

Test Data

|  |  |  |  |
| --- | --- | --- | --- |
| **Object Name** | **Id** | **Name** | **Email** |
| Admin1 | 1 | "Krittika Yothawuth" | "se542115004@vr.camt.info" |

Test Cases

|  |  |  |  |
| --- | --- | --- | --- |
| **Case** | **Description** | **Input** | **Expected Results** |
| **Type: User object** | **Type: admin** |
| 1 | Test with all parameters have input the data | Admin1 | True |
| 2 | Test with input duplicate data | Admin1 | False |
| 3 | Test with empty data | new admin1() | False |
| 4 | Test with null | Null | False |

Unit Test Case (UTC08): testLogin() : Admin

Description

Test testLogin method by getting list of admin information in the database by using admin email and admin password.

Test Function

Public Admin login(email : string, password : string)

Test Data

\*admin1 in database.

|  |  |  |  |
| --- | --- | --- | --- |
| **Object Name** | **Id** | **Email** | **Password** |
| Admin1 | 1 | "se542115004@vr.camt.info" | "1234" |

Test Cases

|  |  |  |  |
| --- | --- | --- | --- |
| **Case** | **Description** | **Input** | **Expected Results** |
| **Type: User object** | **Type: admin** |
| 1 | Test with all parameters have input the data | Admin1 | True |
| 2 | Test with input duplicate data | Admin1 | False |
| 3 | Test with empty data | new admin1() | False |
| 4 | Test with null | Null | False |

Unit Test Case (UTC09): testGetMessage() : List<Message>

Description

Test testGetMessage method by getting full message Test Function

Public getMessage()

Test Data

\*.

Test Cases

Unit Test Case (UTC10):

Unit Test Case (UTC011):

Unit Test Case (UTC12): testGetHistory() : List<History>

Description

Test testGetHistory method by getting list of user information in the database.

Test Function

Public List<History>getHostory()

Test Data

\* history1 , history2 , history3 and history4 is in database

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Object Name** | **Admin ID** | **Time** | **Date** | **Status** |
| History1 | 1 | "08:30:00" | "2015-06-13" | 0 |
| History2 | 2 | "08:45:00" | "2015-08-06" | 1 |
| History3 | 3 | "09:00:00" | "2012-09-25" | 0 |
| History4 | 4 | "09:15:00" | "2000-01-31" | 1 |

Test Cases

|  |  |  |
| --- | --- | --- |
| **Case** | **Description** | **Expected Result** |
| **Type: List of History object** |
| 1 | Test getting list of history | { history1 , history2 , history3 and history4 } |

Unit Test Case (UTC13): testUpdateHistory() : History

Description

Test testUpdateHortory method by inputting history information to the database.

Test Function

Public history updateHistory(history : History)

Test Data

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Object Name** | **Admin ID** | **Time** | **Date** | **Status** |
| History1 | 1 | "08:30:00" | "2015-06-14" | 0 |
| History2 | 2 | "08:30:00" | "2015-06-14" | 0 |
| History3 | 3 | null | "2015-06-14" | 0 |
| History4 | 4 | "08:30:00" | null | 0 |

Test Cases

|  |  |  |  |
| --- | --- | --- | --- |
| **Case** | **Description** | **Input** | **Expected Results** |
| **Type: History object** | **Type: history** |
| 1 | Test with all parameters have input the AdminID, Time, Date and Status | History1 | True |
| 2 | Test with input duplicate data | History1 | False |
| 3 | Test with null time | History3 | False |
| 4 | Test with null date | History4 | False |
| 5 | Test with empty data | new History() | False |
| 6 | Test with null | Null | False |

Unit Test Case (UTC14): test