|  |  |
| --- | --- |
|  | Description |
| /usermanagementservice/authenticateUser | To authenticate a user.  POST  INPUT: JSON of UserInfo object with username and password set.  OUTPUT:  JSON of STATUS class  If authenticated and user type is customer then OUTPUT:  {  "code": 110,  "message": "Authentication Success"  }  If authenticated and user type is admin then OUTPUT:  {  "code": 100,  "message": "Authentication Success"  }  If user with the given email doesn’t exists:  OUTPUT:  {  "code": 103,  "message": "User doesnot exits"  }  If user exists but invalid password, then output:  {  "code": 104,  "message": "Invalid Password"  } |
| /registrymanagement/getallregistry/{user-ID} | To get all registries of the given userID.  GET  INPUT: -  OUTPUT: JSON of list of registries  [  {  "registryID": 900,  "registryName": "MyPublicRegistry",  "registryType": 0,  "userID": 124  },  {  "registryID": 901,  "registryName": "MyPrivateRegistry",  "registryType": 1,  "userID": 124  },  {  "registryID": 903,  "registryName": "My\_NEW\_PublicRegistry",  "registryType": 1,  "userID": 124  }  ] |
| /usermanagementservice/updateUserProfileInfo/ | To update profile information or password  POST  INPUT:  JSON of UserInfo  {  "email": "test1@gmail.com",  "password": "mynewwpassword",  "phoneNum": 123456789,  "userID": 124,  "userType": 1,  "username": "User123RAD1"  }  OUTPUT:  JSON of UserInfo  {  "email": "test1@gmail.com",  "password": "mynewwpassword",  "phoneNum": 123456789,  "userID": 124,  "userType": 1,  "username": "User123RAD1"  } |
| /usermanagementservice/createNewUser | To add a new user to DB.  POST:  INPUT:  JSON of UserInfo  {  "email": "dbcheck1@gmail.com",  "password": "updatedPass",  "phoneNum": 123456789,  "userType": 1,  "username": "User123RAD1"  }  OUTPUT: JSON of UserInfo  {  "email": "dbcheck1@gmail.com",  "password": "updatedPass",  "phoneNum": 123456789,  "userID": 135,  "userType": 1,  "username": "User123RAD1"  } |
| usermanagementservice/getprofileinformation/{user-ID}  Eg:  usermanagementservice/getprofileinformation/135 | To get userInfo for the profile Info page.  GET  INPUT: -  OUTPUT: JSON of UserInfo  {  "email": "dbcheck1@gmail.com",  "password": "updatedPass",  "phoneNum": 123456789,  "userID": 135,  "userType": 1,  "username": "User123RAD1"  } |
| /registrymanagement/createRegistry | To create a new registry.  POST  INPUT: JSON of new registry  {  "registryID": 900,  "registryName": "UpdatedMyPublicRegistry",  "registryType": 0,  "userID": 124  }  OUTPUT:  JSON of status  {  "code": 100,  "message": "SUCCESS"  }  or if you try to create a new registry with same name under same userID, you will get the following response.  JSON of status  {  "code": 102,  "message": "Registry already exits"  } |
| /productmangement/getallproducts/ | To get all the list of Products for admin page.  GET  INPUT: -  OUTPUT:  JSON of Product  [  {  "brandID": 1,  "brandname": "",  "category": "Electronics",  "categoryID": 1,  "certification": 1,  "imageURL": "",  "price": 40,  "productID": 900,  "productName": "PEN",  "rating": 4  },  {  "brandID": 1,  "brandname": "",  "category": "Electronics",  "categoryID": 1,  "certification": 1,  "imageURL": "",  "price": 1400,  "productID": 902,  "productName": "IPHONE",  "rating": 5  }  ] |
| /productmangement/getallbrands | To get all brands. Used in admin page -> add product form.  GET  INPUT: -  OUTPUT:  JSON of brand  [  {  "brandID": 1,  "brandName": "Nike"  },  {  "brandID": 2,  "brandName": "Wildcraft"  }  ] |
| /productmangement/getallcategories | To get all categories. Used in admin page -> add product form.  GET  INPUT: -  OUTPUT:  JSON of Category  [  {  "categoryID": 1,  "categoryName": "Electronics"  },  {  "categoryID": 2,  "categoryName": "Kitchen"  },  {  "categoryID": 3,  "categoryName": "Fashion"  }  ] |
| productmangement/addProduct | To add a product. Used in admin page  POST  INPUT: JSON of product.  BrandID and CategoryID must be set.  BrandName and CategoryName can be empty.  ProductID is not set in the input, it is auto –generated from DB.  {  "brandID": 1,  "brandname": "",  "category": "Electronics",  "categoryID": 1,  "certification": 1,  "imageURL": "",  "price": 1400,  "productName": "MAC",  "rating": 5  }  OUTPUT:  JSON of Status  {  "code": 100,  "message": "SUCCESS"  } |
| /productmangement/deleteProduct/{product-ID} | To delete a product, called from admin page.  GET  INPUT: -  OUPUT: JSON of status.  {  "code": 100,  "message": "Success"  } |
| productregistrymangement/addProductToRegistry/{registryID}/{productID}  productregistrymangement/addProductToRegistry/909/905 | To add product to registry  GET  INPUT: -  OUTPUT: JSON of status  {  "code": 100,  "message": "SUCCESS"  } |
| /productregistrymangement/deleteProductFromRegistry/{registryID}/{productID}  /productregistrymangement/deleteProductFromRegistry/909/905 | To delete a product from registry.  INPUT: -  OUTPUT: JSON of status  {  "code": 100,  "message": "SUCCESS"  } |
| productregistrymangement/getProductsFromRegistry/{RegistryID}  productregistrymangement/getProductsFromRegistry/903 | To get all products from registry which was added to a registry.  INPUT: -  OUTPUT: JSON of LIST of Products  [  {  "brandID": 1,  "brandname": "",  "category": "Electronics",  "categoryID": 1,  "certification": 1,  "imageURL": "",  "price": 40,  "productID": 900,  "productName": "PEN",  "rating": 4  },  {  "brandID": 1,  "brandname": "",  "category": "Electronics",  "categoryID": 1,  "certification": 1,  "imageURL": "",  "price": 1400,  "productID": 902,  "productName": "IPHONE",  "rating": 5  }  ] |
| registrysharingservice/shareprivateregistry/{regID}/{regOwnerUserID}/{buyerUSerID}  registrysharingservice/shareprivateregistry/901/124/test23@gmail.com | To share a registry with user.  GET  INPUT: -  OUTPUT: -  {  "code": 100,  "message": "SUCCESS"  }  If you try to share it to same user again.  {  "code": 200,  "message": "UNABLE TO ADD ENTRY TO DB"  }  for invalid user like  /registrysharingservice/shareprivateregistry/901/124/test2ghfgffhhj3@gmail.com  {  "code": 300,  "message": "USER DOESNOT EXITS"  } |
| /registrymanagement/delete/{registryID}  /registrymanagement/delete/912 | To delete a registry.  GET  INPUT: -  OUTPUT:  JSON of Status  {  "code": 100,  "message": "Success"  }  {  "code": 200,  "message": "Failed"  } |
| /registrymanagement/makepublic/{registryID}  /registrymanagement/makepublic/910 | Make a registry public.  GET  INPUT: -  OUTPUT: -  JSON of Status  {  "code": 100,  "message": "SUCCESS"  } |
| registrymanagement/getallsharedregistry/{userID}  registrymanagement/getallsharedregistry/127 | To get all shared registries. (Includes private registries which where personally shared with you and other public registries which is not owned by you)  GET  INPUT: -  OUTPUT: Json of list of registry  [  {  "registryID": 900,  "registryName": "MyPublicRegistry",  "registryType": 1,  "userID": 124  },  {  "registryID": 901,  "registryName": "MyPrivateRegistry",  "registryType": 1,  "userID": 124  },  {  "registryID": 903,  "registryName": "My\_NEW\_PublicRegistry",  "registryType": 1,  "userID": 124  },  {  "registryID": 909,  "registryName": "MyPublicRegistry",  "registryType": 0,  "userID": 124  }  ] |
| productregistrymangement/buyerMapping/{RegistryID}/{ProductID}/{BuyerUserID}  productregistrymangement/buyerMapping/903/900/129 | To assign a product to self from shared registry  POST  INPUT: -  OUTPUT: -  Json of status  {  "code": 100,  "message": "SUCCESS"  } |
| /productregistrymangement/getProductsAssignedToSelf/{regID}/{buyerUserID}  /productregistrymangement/getProductsAssignedToSelf/903/129 | To get the value of list of products already assigned to self in a particular registry.  GET.  INPUT: -  OUTPUT:  Json of list of Product  [  {  "brandID": 1,  "brandname": "",  "category": "Electronics",  "categoryID": 1,  "certification": 1,  "imageURL": "",  "price": 40,  "productID": 900,  "productName": "PEN",  "rating": 4  },  {  "brandID": 1,  "brandname": "",  "category": "Electronics",  "categoryID": 1,  "certification": 1,  "imageURL": "",  "price": 1400,  "productID": 902,  "productName": "IPHONE",  "rating": 5  }  ] |