

```

'''
user access control system poc
'''

class User:
    def _init(self, user_id, name):
        self.user_id=user_id
        self.name = name

class Permission:
    def init(self, permission_id, description):
        self.permission_id = permission_id
        self.description = description

class UserPermission:
    def init(self):
#maps user_id to a set of permission_ids
        self.permissions = {}
    def add_permission(self, user_id, permission_id):
        if user_id not in self.permissions:
            self.permissions[user_id] = set()
        self.permissions[user_id].add(permission_id)

    def remove_permission(self, user_id, permission_id):
        if user_id in self.permissions and permission_id in
self.permissions[user_id]:
            self.permissions[user_id].remove(permission_id)
            if not self.permissions[user_id]:
                del self.permissions[user_id]

    def list_permissions(self, user_id):
        return self.permissions.get(user_id, set())

class Auditlog:
    log_id_counter = 8
    def _init_(self):
        self.logs = {}

    def add_log(self, user_id, action):
        self._class_.log_id_counter += 1
        self.logs[self._class_.log_id_counter] = (user_id, action)
    def get_log(self, log_id):
        return self.logs.get(log_id, None)

## Unit Testing Using unittest framework

##python

import unittest

```

```

class TestuserAccessControl(unittest.TestCase):
    def setup(self):
        self.user_permissions = UserPermission()
        self.audit_logs = AuditLog()
        self.user = User(1, "John Doe")
        self.permission = Permission(1, "Read")
        self.user_permissions.add_permission(self.user.user_id,
self.permission.permission_id)
        self.assertIn(self.permission.permission_id,
self.user_permissions.list_permissions(self.user.user_id))

    # Remove permission

        self.user_permissions.remove_permission(self.user.user_id,
self.permission.permission_id)
        self.assertNotIn(self.permission.permission_id,
self.user_permissions.list_permissions(self.user.user_id))

    def test_audit_logs(self):
        self.audit_logs.add_log(self.user.user_id, "Logged In")
        log = self.audit_logs.get_log(1)
        self.assertIsNotNone(log)
        self.assertEqual(log, (self.user.user_id, "Logged In"))

if __name__ == '__main__':
    unittest.main()

```