

**LOG3430 –Méthodes et Tests**

**Hiver 2021**

**TP No. [1]**

**Groupe [ 1]**

**[1947025] – [Cassy Charles]**

**[] – [Khadija Rhekik]**

**[14 Février 2021]**

# La couverture de jeux de tests initiale.

Test\_crud.py

Text

Description automatically generated

Test\_Vocabulary\_Creator.py

A screen shot of a computer

Description automatically generated with low confidence

Test\_email\_analyzer.py

A screen shot of a computer

Description automatically generated with low confidence

# La couverture du jeu des tests après les tests.

Test\_crud.py

Text

Description automatically generated

9 tests ont été ajoutés en plus de ceux fournis.

Tests ajoutés :

def test\_add\_new\_user\_Returns\_false\_to\_invalid\_email\_to\_modify\_users\_file(  
 self, mock\_modify\_users\_file, mock\_modify\_groups\_file, mock\_read\_users\_file  
):  
 crud=CRUD()  
 email="test"  
 date="2003-09-13"  
  
 self.assertFalse(crud.add\_new\_user(email,date))

Ce test a été ajouté pour vérifier que le courriel de l’utilisateur est conforme au système, sinon il retourne faux

@patch("crud.CRUD.modify\_users\_file")  
@patch("crud.CRUD.read\_users\_file")  
def test\_update\_users\_Passes\_correct\_value\_of\_Trust\_to\_modify\_users\_file(  
 self, mock\_read\_users\_file, mock\_modify\_users\_file  
):  
 mock\_read\_users\_file.return\_value = self.users\_data  
  
 crud=CRUD()  
 user\_id="2"  
 field="Trust"  
 data=60  
  
 crud.update\_users(user\_id,field,data)  
 mock\_modify\_users\_file.assert\_called\_once\_with(self.users\_data)

Ce test a été ajouté pour vérifier que si le field est Trust, la donnée de l’utilisateur est bien remplacée dans le tableau

@patch("crud.CRUD.modify\_users\_file")  
 @patch("crud.CRUD.read\_users\_file")  
 def test\_update\_users\_Passes\_correct\_value\_of\_name\_to\_modify\_users\_file(  
 self, mock\_read\_users\_file, mock\_modify\_users\_file  
):  
 mock\_read\_users\_file.return\_value = self.users\_data  
  
 crud=CRUD()  
 user\_id="2"  
 field="name"  
 data="Test@gmail.com"  
  
 crud.update\_users(user\_id,field,data)  
 mock\_modify\_users\_file.assert\_called\_once\_with(self.users\_data)

Ce test couvre la branche quand le field suggéré par l’utilisateur est name et change cette valeur dans le users\_data

@patch("crud.CRUD.modify\_users\_file")  
 @patch("crud.CRUD.read\_users\_file")  
 def test\_update\_users\_Passes\_correct\_value\_of\_groups\_modify\_users\_file(  
 self, mock\_read\_users\_file, mock\_modify\_users\_file  
):  
 mock\_read\_users\_file.return\_value = self.users\_data  
  
 crud=CRUD()  
 user\_id="2"  
 field="Groups"  
 data=[]  
  
 crud.update\_users(user\_id,field,data)  
 mock\_modify\_users\_file.assert\_called\_once\_with(self.users\_data)

Ce test couvre la branche quand le field suggéré par l’utilisateur est Groups et change cette valeur dans le users\_data

@patch("crud.CRUD.modify\_users\_file")  
@patch("crud.CRUD.read\_users\_file")  
def test\_update\_users\_Passes\_correct\_value\_of\_date\_first\_seen\_to\_modify\_users\_file(  
 self, mock\_read\_users\_file, mock\_modify\_users\_file  
):  
 mock\_read\_users\_file.return\_value = self.users\_data  
  
 crud=CRUD()  
 user\_id="1"  
 field="Date\_of\_first\_seen\_message"  
 data="2000-09-01"  
  
 crud.update\_users(user\_id,field,data)  
 mock\_modify\_users\_file.assert\_called\_once\_with(self.users\_data)

Ce test couvre la branche quand le field suggéré par l’utilisateur est Date\_of\_first\_Seen\_message et change cette valeur dans le users\_data. Ce test aussi rentre dans la fonction convertDate et la couvre

@patch("crud.CRUD.modify\_users\_file")  
 @patch("crud.CRUD.read\_users\_file")  
 def test\_update\_users\_Passes\_correct\_value\_of\_date\_last\_seen\_to\_modify\_users\_file(  
 self, mock\_read\_users\_file, mock\_modify\_users\_file  
):  
 mock\_read\_users\_file.return\_value = self.users\_data  
  
 crud=CRUD()  
 user\_id="2"  
 field="Date\_of\_last\_seen\_message"  
 data="2022-09-01"  
  
 crud.update\_users(user\_id,field,data)  
 mock\_modify\_users\_file.assert\_called\_once\_with(self.users\_data)

Ce test couvre la branche quand le field suggéré par l’utilisateur est Date\_of\_last\_Seen\_message et change cette valeur dans le users\_data.

@patch("crud.CRUD.modify\_groups\_file")  
 @patch("crud.CRUD.read\_groups\_file")  
 def test\_update\_groups\_Passes\_correct\_value\_to\_modify\_groups\_file(  
 self, mock\_read\_groups\_file, mock\_modify\_groups\_file  
):  
 mock\_read\_groups\_file.return\_value = self.groups\_data  
  
 crud=CRUD()  
 group\_id=2  
 field="Trust"  
 data=60  
  
 crud.update\_groups(group\_id,field,data)  
 mock\_modify\_groups\_file.assert\_called\_once\_with(self.groups\_data)

Ce test couvre la branche quand le field suggéré par l’utilisateur est Trust et change cette valeur dans le groups\_data.

@patch("crud.CRUD.modify\_groups\_file")  
 @patch("crud.CRUD.read\_groups\_file")  
 def test\_update\_groups\_Passes\_correct\_value\_of\_name\_to\_modify\_users\_file(  
 self, mock\_read\_groups\_file, mock\_modify\_groups\_file  
):  
 mock\_read\_groups\_file.return\_value = self.groups\_data  
  
 crud=CRUD()  
 group\_id=2  
 field="name"  
 data="Test"  
  
 crud.update\_groups(group\_id,field,data)  
 mock\_modify\_groups\_file.assert\_called\_once\_with(self.groups\_data)

Ce test couvre la branche quand le field suggéré par l’utilisateur est name et change cette valeur dans le groups\_data.

@patch("crud.CRUD.modify\_groups\_file")  
@patch("crud.CRUD.read\_groups\_file")  
def test\_update\_groups\_Passes\_correct\_value\_of\_list\_of\_members\_to\_modify\_users\_file(  
 self, mock\_read\_groups\_file, mock\_modify\_groups\_file  
):  
 mock\_read\_groups\_file.return\_value = self.groups\_data  
  
 crud=CRUD()  
 group\_id=2  
 field="List\_of\_members"  
 data=[]  
  
 crud.update\_groups(group\_id,field,data)  
 mock\_modify\_groups\_file.assert\_called\_once\_with(self.groups\_data)

Ce test couvre la branche quand le field suggéré par l’utilisateur est List\_of\_members et change cette valeur dans le groups\_data.