**Test1:**

Test case for canRecieveMessage() function.

Consider a recipient is allowed to be added to a message. If the current user doesn’t have permission to modify recipients, the supplied messages should be added to failed messages. In this case the current code returns true in canRecieveMessage function and the message will be added to the successfulMessages array.

**Expected output** is failedmessages size should return 2

**Actual output** is failedmessages size 0

//Create test users

User recipient = new User (‘user123’);

User currentUser = new User(‘user456’);

//Create test messages

Message Message1 = new Message (“Test1”);

Message Message2 = new Message (“Test2”);

// setHasPermissionToModifyRecipients for users

recipient.setHasPermissionToModifyRecipients(false);

currentUser.setHasPermissionToModifyRecipients(false);

Message1.setAddRecipientAllowed(recipient, true)

Message2.setAddRecipientAllowed(recipient, true)

ArrayList<Message> messages = new ArrayList<Message>();

messages.add(Message1);

messages.add(Message2);

assertEquals(messages.size(),2);

ArrayList<Message> result = rec.addRecipientToAllMessages(messages, currentUser, recipient)

//Failed messages should have size 2

assertEquals(results.size(),2);

**Test 2:**

Test for addRecipient() function : This function is not called when the successfulMessages array is not empty.

Expected: Sending two successful messages should add recipient to the messages

Actual output: Sending two successful messages doesn’t add recipient to the messages

//Create test users

User recipient = new User (‘user123’);

User currentUser = new User(‘user456’);

//Create test messages

Message Message1 = new Message (“Test1”);

Message Message2 = new Message (“Test2”);

// setHasPermissionToModifyRecipients for users

recipient.setHasPermissionToModifyRecipients(true);

currentUser.setHasPermissionToModifyRecipients(true);

Message1.setAddRecipientAllowed(recipient, true)

Message2.setAddRecipientAllowed(recipient, true)

ArrayList<Message> messages = new ArrayList<Message>();

messages.add(Message1);

messages.add(Message2);

assertEquals(messages.size(),2);

ArrayList<Message> result = rec.addRecipientToAllMessages (messages, currentUser, recipient)

//Assuming a function to check the number of the messages for a given recipient //(messagesWithRecipient ())

//Below function should pass. Due to the bug in the code it fails

assertEquals(messagesWithRecipient(recipient).size(),2);

**Test 3:**

Test for addRecipient(currentUser, successfulMessages)

//Create test users

User recipient = new User (‘user123’);

User currentUser = new User(‘user456’);

//Create test messages

Message Message1 = new Message (“Test1”);

Message Message2 = new Message (“Test2”);

// setHasPermissionToModifyRecipients for users

recipient.setHasPermissionToModifyRecipients(true);

currentUser.setHasPermissionToModifyRecipients(true);

Message1.setAddRecipientAllowed(recipient, true)

Message2.setAddRecipientAllowed(recipient, true)

ArrayList<Message> messages = new ArrayList<Message>();

messages.add(Message1);

messages.add(Message2);

assertEquals(messages.size(),2);

ArrayList<Message> result = rec.addRecipientToAllMessages(messages, currentUser, recipient)

//Assuming a function ‘verifyPassedRecipientToaddRecypient()’ to verify that recipient is passed to //‘addRecipient’ function

assertEquals(verifyPassedRecipientToaddRecypient(), true);