***Three logic bugs:***

* **Return statement** in the method - canReceiveMessage(Message message, User currentUser, User recipient)
  + Return statement should be false, if the conditions are not met
* **Condition statement** in the If loop
  + If(currentUser.equals(recipient) || currentUser.getHasPermissionToModifyRecipients())
  + In the above- currentUser.equals(recipient) Condition is not required
* **Condition statement** when the user has some successful messages, we add the user to recipientmnager. We need to add only when the User has some successful messages.
  + If(successfulMessages.isEmpty()) – This should be if(!successfulMessages.isEmpty())

***Test Code:***

Test code need to test various inputs (positive and negative) of the logics, It is only one case user have tested.

Also, Test cases has to be designed the way functionality is expected, AssertEquals statements only give us an idea on the messages.

***Tests to test the 3 bugs:***

//Validating the return statement

public TestAddRecipients(){

User user = new User("admin");

User userRecipient = new User("recipientUser");

user.setHasPermissionToModifyRecipients(false);

Message goodMessage = new Message("Test 1");

goodMessage.setAddRecipientAllowed(recipientUser, true);

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

messages.add(goodMessage);

//Assert that we pass two Messages into the function

asserEquals(messages.size(), 1);

ArrayList<Message> result = rec.addRecipientToAllMessages(messages, recipientUser, user);

//Assert that only one message failed to be updated

asserEquals(result.size(), 0);

}

//Validating the Condition in the if loop

public TestAddRecipients(){

User user = new User("admin");

user.setHasPermissionToModifyRecipients(false);

Message goodMessage = new Message("Test 1");

goodMessage.setAddRecipientAllowed(admin, true);

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

messages.add(goodMessage);

//Assert that we pass two Messages into the function

asserEquals(messages.size(), 1);

ArrayList<Message> result = rec.addRecipientToAllMessages(messages, user, user);

//Assert that only one message failed to be updated

asserEquals(result.size(), 0);

}

//Validate the recipientManager object

public TestAddRecipients(){

User user = new User("admin");

user.setHasPermissionToModifyRecipients(true);

Message goodMessage = new Message("Test 1");

Message badMessage = new Message("Test 2");

goodMessage.setAddRecipientAllowed(user, false);

badMessage.setAddRecipientAllowed(user, false);

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

messages.add(goodMessage);

messages.add(badMessage);

//Assert that we pass two Messages into the function

asserEquals(messages.size(), 2);

ArrayList<Message> result = rec.addRecipientToAllMessages(messages, user, user);

//Assert that only one message failed to be updated

asserEquals(result.size(), 0);

//Validate the recipientManager objects with the existence of the users

//Verify the recipientManagerObject and "admin" shou;dn't be present

}