

CS352 Software Engineering II

Software Testing Report

Chat Server

Team “To Be”

- | | | | |
|--|----------------------|-----------------|--------------|
| 1. Ahmed Hamed Sayed Hamed | (Team Leader) | 20110046 | CS G1 |
| 2. Eslam Alaa Eldin Ibrahim | | 20110022 | CS G1 |
| 3. Ashraf Mohey | | 20110114 | CS G1 |
| 4. Abdallah Mohamed Abdelgafour | | 20110230 | CS G1 |

TA

Sabah



CS352: Phase 3 – C (Dr. Mohammad El-Ramly)

Development Plan

Introduction:

- This document contains explaining for the **unit testing** process for the biggest class (**ChatServer Class**) using JUnit tool and TestNG tool, including screenshots from code to clarify test cases are selected by tester, Unit testing codes by tools is attached in same folder with testing report.
- This document contains explaining for Behavior Testing Using JBehave tool.
- Also document contains explaining for using BMD analysis tool on whole code.

Unit Test Cases and Results:

I. Definitions:

```
@Before
public void setUp() throws Exception {
    chatServer_ = new ChatServer().getInstance();
}

@After
public void tearDown() throws Exception {
    chatServer_ = null;
}
```

1. setup Function and tearDown Function is called automatic when tester runs testing code.

2. setUp Function:

- Function is called before each function which tester wants to test to create and initializing constructor from class which hold function.

3. tearDown Function:

- Function is called after each function which tester wants to test to delete constructor from class which hold function.

II. ChatServer Class:

1. addUser Function Unit Testing:

```
@Test
public void testAddUser() {
    GeneralUser user = new GeneralUser();
    user.setName("mohamed");
    user.setPassword("1234");
    int sizeBeforeTest1 = chatServer_.getUsers().size();
    chatServer_.addUser(user);
    int sizeAfterTest1 = chatServer_.getUsers().size();
    assertTrue(sizeBeforeTest1 + 1 == sizeAfterTest1);

    int sizeBeforeTest2 = chatServer_.getUsers().size();
    chatServer_.addUser(user);
    int sizeAfterTest2 = chatServer_.getUsers().size();
    assertTrue(sizeBeforeTest2 == sizeAfterTest2);
}
```

- Test cases for these functions **passed**.



CS352: Phase 3 – C (Dr. Mohammad El-Ramly)

Development Plan

2. **addRoomAdmin** Function Unit Testing:

```
@Test
public void testAddRoomAdmin() {
    AdminUser admin = new AdminUser();

    List<String> permissions = new ArrayList<String>();
    admin.setID(1);
    admin.setName("mohamed");
    admin.setPassword("1234");
    permissions.add("RU");
    permissions.add("RR");
    int permissionsSizeBeforeTest1 = admin.getPermissions().size();
    chatServer_.addRoomAdmin(admin, permissions);
    int permissionsSizeAfterTest1 = admin.getPermissions().size();
    assertTrue(permissionsSizeBeforeTest1 + 2 == permissionsSizeAfterTest1);

    permissions = new ArrayList<String>();
    permissions.add("OR");
    int permissionsSizeBeforeTest2 = admin.getPermissions().size();
    chatServer_.addRoomAdmin(admin, permissions);
    int permissionsSizeAfterTest2 = admin.getPermissions().size();
    assertTrue(permissionsSizeBeforeTest2 + 1 == permissionsSizeAfterTest2);
}
```

- Test cases for these functions **passed**.

3. **removeUser** Function Unit Testing:

```
@Test
public void testRemoveUser() {
    GeneralUser user = new GeneralUser();
    user.setID(1);
    user.setName("mohamed");
    user.setPassword("1234");
    chatServer_.addUser(user);
    int sizeBeforeTest1 = chatServer_.getUsers().size();
    chatServer_.removeUser(user.getID());
    int sizeAfterTest1 = chatServer_.getUsers().size();
    assertTrue(sizeBeforeTest1 == sizeAfterTest1 + 1);

    int sizeBeforeTest2 = chatServer_.getUsers().size();
    chatServer_.removeUser(user.getID());
    int sizeAfterTest2 = chatServer_.getUsers().size();
    assertTrue(sizeBeforeTest2 == sizeAfterTest2);
}
```

- Test cases for these functions **passed**.



CS352: Phase 3 – C (Dr. Mohammad El-Ramly)

Development Plan

4. removeAllUser Function Unit Testing:

```
@Test
public void testRemoveAllUser() {
    IRoom room = new AbstractRoom();
    IUser user = new AbstractUser();
    user.setID(1);
    user.setName("mohamed");
    user.setPassword("1234");
    room.setID(1);
    room.setTitle("room");
    room.setDescription("general");
    room.addUser(user);
    chatServer_.getRooms().add(room);
    chatServer_.removeAllUser(room.getID());
    int sizeAfterTest1 = chatServer_.getRooms().get(0).getUsers().size();
    assertTrue(sizeAfterTest1 == 0);

    chatServer_.removeAllUser(room.getID());
    int sizeAfterTest2 = chatServer_.getRooms().get(0).getUsers().size();
    assertTrue(sizeAfterTest2 == 0);
}
```

- Test cases for these functions failed.

5. removeRoom Function Unit Testing:

```
@Test
public void testRemoveRoom() {
    IRoom room = new AbstractRoom();
    room.setID(1);
    room.setTitle("room");
    room.setDescription("general");
    chatServer_.getRooms().add(room);
    chatServer_.removeRoom(room.getID());
    int sizeAfterTest1 = chatServer_.getRooms().size();
    assertTrue(sizeAfterTest1 == 0);

    chatServer_.removeRoom(room.getID());
    int sizeAfterTest2 = chatServer_.getRooms().size();
    assertTrue(sizeAfterTest2 == 0);
}
```

- Test cases for these functions passed.



CS352: Phase 3 – C (Dr. Mohammad El-Ramly)

Development Plan

6. **getRooms** Function Unit Testing:

```
@Test
public void testGetRooms() {
    IRoom room = new AbstractRoom();
    room.setID(1);
    room.setTitle("room");
    room.setDescription("general");
    int sizeBeforeTest = chatServer_.getRooms().size();
    chatServer_.getRooms().add(room);
    int sizeAfterTest = chatServer_.getRooms().size();
    assertTrue(sizeBeforeTest + 1 == sizeAfterTest);
}
```

- Test cases for these functions **passed**.

7. **getUsers** Function Unit Testing:

```
@Test
public void testGetUsers() {
    IUser user = new AbstractUser();
    user.setID(1);
    user.setName("mohamed");
    user.setPassword("1234");
    int sizeBeforeTest = chatServer_.getUsers().size();
    chatServer_.getUsers().add(user);
    int sizeAfterTest = chatServer_.getUsers().size();
    assertTrue(sizeBeforeTest + 1 == sizeAfterTest);
}
```

- Test cases for these functions **passed**.

8. **joinRoom** Function Unit Testing:

```
@Test
public void testJoinRoom() {
    IRoom room = new AbstractRoom();
    IUser user = new AbstractUser();
    user.setID(1);
    user.setName("mohamed");
    user.setPassword("1234");
    chatServer_.getUsers().add(user);
    room.setID(1);
    room.setTitle("room");
    room.setDescription("general");
    chatServer_.getRooms().add(room);
    int sizeBeforeTest = chatServer_.getRooms().get(0).getUsers().size();
    chatServer_.joinRoom(room.getID(), user.getID());
    int sizeAfterTest = chatServer_.getRooms().get(0).getUsers().size();
    assertTrue(sizeBeforeTest + 1 == sizeAfterTest);
}
```

- Test cases for these functions **passed**.



CS352: Phase 3 – C (Dr. Mohammad El-Ramly)

Development Plan

9. createRestirctedRoom Function Unit Testing:

```
@Test
public void testCreatRestirctedRoom() {
    IUser user = new AbstractUser();
    user.setID(1);
    user.setName("mohamed");
    user.setPassword("1234");
    List<IUser> allowdUsers = new ArrayList<IUser>();
    allowdUsers.add(user);

    int sizeBeforeTest1 = chatServer_.getRooms().size();
    chatServer_.creatRestirctedRoom("room", "general", allowdUsers);
    int sizeAfterTest1 = chatServer_.getRooms().size();
    assertTrue(sizeBeforeTest1 + 1 == sizeAfterTest1);

    int sizeBeforeTest2 = chatServer_.getRooms().size();
    chatServer_.creatRestirctedRoom("room", "general", allowdUsers);
    int sizeAfterTest2 = chatServer_.getRooms().size();
    assertTrue(sizeBeforeTest2 == sizeAfterTest2);
}
```

- Test cases for these functions failed.

10. createGenralRoom Function Unit Testing:

```
@Test
public void testCreateGenralRoom() {
    IUser user = new AbstractUser();
    user.setID(1);
    user.setName("mohamed");
    user.setPassword("1234");

    int sizeBeforeTest1 = chatServer_.getRooms().size();
    chatServer_.createGenralRoom("room", "general", (AbstractUser) user);
    int sizeAfterTest1 = chatServer_.getRooms().size();
    assertTrue(sizeBeforeTest1 + 1 == sizeAfterTest1);

    int sizeBeforeTest2 = chatServer_.getRooms().size();
    chatServer_.createGenralRoom("room", "general", (AbstractUser) user);
    int sizeAfterTest2 = chatServer_.getRooms().size();
    assertTrue(sizeBeforeTest2 == sizeAfterTest2);
}
```

- Test cases for these functions passed.



CS352: Phase 3 – C (Dr. Mohammad El-Ramly)

Development Plan

11. leaveRoom Function Unit Testing:

```
@Test
public void testLeaveRoom() {
    IRoom room = new AbstractRoom();
    IUser user = new AbstractUser();
    user.setID(1);
    user.setName("mohamed");
    user.setPassword("1234");
    room.setID(1);
    room.setTitle("room");
    room.setDescription("general");
    room.addUser(user);
    chatServer_.getRooms().add(room);

    int sizeBeforeTest1 = room.getUsers().size();
    chatServer_.leaveRoom(room.getID(), user.getID());
    int sizeAfterTest1 = room.getUsers().size();
    assertTrue(sizeBeforeTest1 == sizeAfterTest1 + 1);

    int sizeBeforeTest2 = room.getUsers().size();
    chatServer_.leaveRoom(room.getID(), user.getID());
    int sizeAfterTest2 = room.getUsers().size();
    assertTrue(sizeBeforeTest2 == sizeAfterTest2);
}
```

- Test cases for these functions failed.

12. getInstance Function Unit Testing:

```
@Test
public void testGetInstance() {
    ChatServer chatServerTesting = new ChatServer().getInstance();
    if (chatServerTesting == null)
        assertTrue(false);
}
```

- Test cases for these functions passed.



CS352: Phase 3 – C (Dr. Mohammad El-Ramly)

Development Plan

13. **blockUser** Function Unit Testing:

```
@Test
public void testBlockUser() {
    GeneralUser user = new GeneralUser();
    GeneralUser blockedUser = new GeneralUser();
    user.setID(1);
    user.setName("mohamed");
    user.setPassword("1234");
    chatServer_.getUsers().add(user);
    blockedUser.setID(2);
    blockedUser.setName("eslam");
    blockedUser.setPassword("0000");
    chatServer_.getUsers().add(blockedUser);
    int sizeBeforeTest = user.getBlockedUsers().size();
    chatServer_.blockUser(user.getID(), blockedUser.getID());
    int sizeAfterTest = user.getBlockedUsers().size();
    assertTrue(sizeBeforeTest + 1 == sizeAfterTest);
}
```

- Test cases for these functions **passed**.

Behavior Testing and Results:

- Codes for this process attach in same folder with testing report.

BMD Testing and Results:

I. Definitions:

- BMD is an analysis tool to find Violations in code. We run tool on whole code and get lots of Violations.
- Pictures show results by BMD tool attach with report in same folder.
- And also attach with pictures txt file explain places of Violations in code.

Ownership Report:

Item	Owner
All Report	<i>Eslam and Abdallah</i>