Testing Plan

- UserTest - The User class encapsulates getters for id (username), password, and status.
    - Constructor - Constructs a user and tests the getters. The default status should be Offline
    - setStatus - Tests changing the status of the user to Online, Away, and Busy
- ChatMessageTest - The ChatMessage class encapsulates information about the contents of the message, including the timestamp and the sender
    - Standard Constructor - Constructs a chat message with the current time and the specified contents, status, and sender, testing the getters.
    - Storage Constructor - Constructs a chat message with the “Delivered” status at the given time with the specified content and sender, testing the getters
    - File Constructor - Constructs a chat message based on the information in the file storage, testing the getters
    - setStatus - Tests changing the message status to Delivered
- ChatHistoryTest - The Chat History encapsulates a collection of chat messages
    - Constructor - Constructs a chat message history, testing that it is empty
    - addMessage - tests adding chat messages to the object and ensuring they are accessible through the get method
    - getMessagesFromUser - tests retrieving all messages from the specified sender
    - setStatus - tests setting the status of a particular chat message by the chat message’s id
- RoomTest - The Room class encapsulates a chat history alongside a set of users
    - Standard Constructor - Constructs a room using a specified list of users and the default title, testing the getters
    - Title Constructor - Constructs a room using a specified list of users and a custom title, testing the getters
    - File Constructor - Constructs a room given a file that stores the room and tests the getters
    - removeUser - Tests removing a certain user from a room by their id
    - addUser - Tests adding a user to a room by their id
- UserStorageTest - The UserStorage class encapsulates the functionality of working with all the users, including the rooms they are in
    - File Constructor - Constructs the user storage based on 2 specific files, testing the getters
    - addUser - Tests adding a new user to the user storage
    - addUserToRoom - Tests adding the user to a particular room, both specified by the id’s
    - removeUserFromRoom - Tests removing a user from a particular room, both specified by the id’s
    - setUserStatus - Tests changing the specified user’s status to Online, Away, and Busy as well as ensuring the default is Offline

- AuthenticatorTest - The Authenticator class encapsulates the functionality of authenticating users
    - testAuthenticate - Constructs a new authenticator based off the default user storage, ensuring the returned message corresponds to the authenticated user and that failed authentication attempts return null
- RoomStorageTest - The RoomStorage class encapsulates the functionality for room operations
    - Constructor - Constructs the room storage based on a predefined directory and tests the room getter
    - addRoom - Tests the functionality of adding a room to the room storage
    - addUser - Tests the functionality of adding a particular user to a particular room, both specified by their id's
    - removeUser - Tests the functionality of adding a particular user to a particular room, both specified by their id's
    - setMessageStatus - Tests the functionality of changing the status of a particular message in a particular room, both specified by their id's
- LoggerTest - The Logger handles the simultaneous operations of the user and room storages
    - Constructor - Construct the logger based on a room storage and user storage, testing to ensure it is constructed
    - createRoom - Tests creating a room and ensuring the room contains the respective users and the users are in the respective rooms
    - addUserToRoom - Tests adding a user to a room in both storages
    - removeUserFromRoom - Tests removing a user from a room in both storages
- MessageTest - The Message class encapsulates all possible message types and supports a variety of types and stored data
    - Constructor - Tests constructing all of the different types of messages
    - setPassword - Tests setting the password and ensuring it is immutable for the Logout message
    - setLoginStatus - Tests setting the login status and ensuring it is immutable for the Login message
    - setRoomId - Tests setting the room id and ensuring it is immutable for the Login message
    - setUser - Tests setting the user and ensuring it is immutable for the Login message
    - setUserStatus - Tests setting the user status and ensuring it is immutable for the Login message
    - setContents - Tests setting the contents and ensuring it is immutable for the NewChat message
    - setRooms - Tests setting the rooms and ensuring it is immutable for the NewRoom message

How to Run

The tests can be run either by running main in the TestDriver class or by selecting the Test class and running it as a JUnit Test. Additionally, individual tests can also be ran using the latter method. For environments which do not support special functionality for JUnit testing, only the former method is functional.