

**SEN-109 Modern Systems Programming**  
**Assessment Activity SEN-109:00010**  
**Understanding systems programming**

Name: Thanawin Pattanaphol

ID: 01324096

1. In Comms Library, explain the difference between a **CommandStream** and an **EventStream**

The difference between the two are based on the usage of the two streams. **CommandStream** is used for receiving the commands that are being sent by the client (the user) and putting them into asynchronous series of values, also known as a stream.

Meanwhile, **EventStream** has a similar functionality but is used for receiving events that are sent by the server.

2. In Server's **main.rs**, we wrap **room\_manager** with **Arc**. What is the purpose of wrapping **Arc** around **room\_manager**? What would happen if we do not use **Arc** to wrap **room\_manager**?

The purpose of wrapping **Arc** or the Atomic Reference Counter around **room\_manager** is because we want to allow the data (in this case, the variable **room\_manager**) to be shared between threads and to ensure that the allocated data gets deallocated or cleaned up when the last reference to it is dropped.

3. In Server's **room\_manager.rs**, the **chat\_rooms** are represented by a **HashMap** with keys of type **String** and values of type **Arc<Mutex<ChatRoom>>**. You can see example usage of **Mutex** in function **join\_room** and **drop\_user\_session\_handle**. What is the purpose of **Mutex**? and why do the values in **chat\_rooms** **HashMap** must have type **Arc<Mutex<ChatRoom>>** instead of just **ChatRoom**?

The purpose of **Mutex** is to ensure that one thread can access data at any time, it's essentially a lock for a specific piece of data. If a thread needs to read or write, it will need to unlock the **Mutex**'s lock first before doing so.

The values in the **chat\_rooms** **HashMap** are needed to be **Arc<Mutex<ChatRoom>>** type due to the fact that there may be several tasks that will be needed to be done on asynchronously on multiple threads, due to the usage of the **Tokio** library, **Arc** is needed for the data to be accessible on multiple threads; meanwhile, **Mutex** is needed those asynchronous tasks like room joining or drop user session handle to only be done by a thread.

4. In TUI's **main.rs**, there are two **main\_loops** defined for the client application: **state\_store** and **ui\_manager**. Analyze the code and describe the purposes of each **main\_loop**. What are the interactions between **state\_store** and **ui\_manager**? Write a diagram showing communication channels between **state\_store** and **ui\_manager**.

The purposes of each **main\_loop** are as following:

1. **state\_store**

**state\_store** is mainly used to handle actions coming from both the server, such as, server events, and from the UI such as: Send Messages, Selecting Rooms, and Exit and with them, process them to do async operations.

**SEN-109 Modern Systems Programming**  
**Assessment Activity SEN-109:00010**  
**Understanding systems programming**

Name: Thanawin Pattanaphol

ID: 01324096

2. ui\_manager

ui\_manager is mainly used as an App Router for the UI (handling the routing of pages) along with handling the asynchronous state updates, events and interrupts.

