Assignment 94: In which different situations messages get posted into an application message queue?

Messages can be posted to an application's message queue in various situations, typically as a result of user actions, system events, or inter-process communication. Here are some common situations in which messages get posted into an application's message queue:

1. **User Input Events**: When a user interacts with an application's user interface, such as clicking a button, typing into a text box, or moving the mouse, corresponding messages are generated and posted to the application's message queue. These messages include events such as `WM_MOUSEMOVE`, `WM_LBUTTONDOWN`, `WM_KEYDOWN`, etc.

2. **Window Management Events**: Messages related to window management, such as window creation, resizing, moving, activation, and closure, are posted to an application's message queue. These messages include events such as `WM_CREATE`, `WM_SIZE`, `WM_MOVE`, `WM_ACTIVATE`, `WM_CLOSE`, etc.

3. **Timer Events**: If an application sets up timer objects using functions like `SetTimer`, timer expiration events are generated at specified intervals and posted to the application's message queue. These messages include `WM_TIMER` messages.

4. **System Events**: Certain system-wide events, such as power events (e.g., system going to sleep or waking up), session events (e.g., user logon or logoff), display changes, or changes in system settings, are posted to the message queue of all running applications. These messages include events such as `WM_POWERBROADCAST`, `WM_DISPLAYCHANGE`, etc.

5. **Inter-process Communication (IPC)**: In scenarios involving inter-process communication, messages can be sent from one process to another using mechanisms like window messages, named pipes, sockets, or other IPC mechanisms. When a message is received by an application, it is posted to the recipient application's message queue for processing.

6. **Custom Messages**: Applications can define custom messages to facilitate communication between different parts of the application or between different modules within the same process. These custom messages are posted to the application's message queue using functions like `PostMessage` or `SendMessage`.

Overall, messages are posted to an application's message queue in response to various events and actions, allowing the application to process them asynchronously and respond accordingly. This event-driven model forms the basis of user interface programming and event handling in Windows applications.