

## Use Case Scenario:

### **Delete Event**

#### Context:

A user has created an event, but the event has been cancelled.

#### Scenario:

##### *Primary*

1. The user adds a valid event.
2. System adds the event
3. System displays the event in the Event list window
4. User clicks the event in the Event list window.
5. System displays event information in the Event Detail window
6. User clicks “Remove” button
7. System Requests user confirmation
8. User confirms to remove event
9. System Removes event

##### *User Does not confirm removing an event*

8. User cancels removing the event

### **Update Events**

#### Context:

A user adds an event they believe is correct, but realizes afterwards that it is not.

#### Scenario:

##### *Primary*

1. The user adds a valid event.
2. System adds the event
3. System displays the event in the Event list window
4. User clicks the event in the Event list window.
5. System displays event information in the Event Detail window
6. User clicks “Update” Window
7. User inputs all valid information for event
8. User submits information
9. System requests confirmation for updating the event
10. User confirms updating the event
11. System updates the event

##### *User Does not confirm updating an event*

10. User cancels updating the event
11. System discards the updated information
12. System exits the Create Event Interface

*Invalid/Missing Information*

1. User inputs some invalid (incorrect or missing) information for event
2. User submits information
3. System indicates all invalid information
4. System returns user to Create Event Interface