

```
001 import de.maxmaurer.FlashGameCommunicator.*;
002 import de.maxmaurer.FlashGameCommunicator.Events.*;
003
004 setUpButtons();
005 var comm:FlashGameCommunicator = new FlashGameCommunicator();
006 setUpCommListeners();
007 comm.connect("localhost",11111);
008
009 function setUpButtons():void {
010     btnSendName.addEventListener(MouseEvent.CLICK, sendNameHandler);
011     btnSendMessage.addEventListener(MouseEvent.CLICK, sendMessageHandler);
012     btnDisconnect.addEventListener(MouseEvent.CLICK, disconnectHandler);
013     btnCreateRoom.addEventListener(MouseEvent.CLICK, createRoomHandler);
014     btnRefreshRooms.addEventListener(MouseEvent.CLICK, refreshRoomsHandler);
015     btnLeaveRoom.addEventListener(MouseEvent.CLICK, leaveRoomHandler);
016     listRooms.addEventListener(MouseEvent.CLICK, listRoomsHandler);
017     btnJoinRoom.addEventListener(MouseEvent.CLICK, joinRoomHandler);
018     btnUserList.addEventListener(MouseEvent.CLICK, userListHandler);
019     btnUserListRoom.addEventListener(MouseEvent.CLICK, userListRoomHandler);
020     btnRoomInformation.addEventListener(MouseEvent.CLICK, roomInformationHandler);
021 }
022
023 function setUpCommListeners():void {
024     comm.addEventListener(GameEvent.CONNECTED,connected);
025     comm.addEventListener(GameEvent.DISCONNECTED,disconnected);
026     comm.addEventListener(GameEvent.MESSAGE,messageIncoming);
027     comm.addEventListener(GameEvent.SUCCESS,success);
028     comm.addEventListener(GameEvent.FAILED,failed);
029     comm.addEventListener(GameEvent.KICKED,kicked);
030 }
031
032 function connected(evt:GameEvent) {
033     txtUserId.text = "UserId: "+evt.myId;
034     taMessages.text += "Verbindung hergestellt\n";
035     checkEnabled();
036 }
037 function disconnected(evt:GameEvent) {
038     taMessages.text+= "Verbindung wurde beendet\n";
039     checkEnabled();
040 }
041 function messageIncoming(evt:GameEvent) {
042     taMessages.text += evt.senderName+": "+evt.message+"\n";
043 }
044 function success(evt:GameEvent) {
045     if (evt.messageType == FlashGameCommunicator.ACTION_SEND_NAME) {
046         taMessages.text += "Name wurde gesetzt\n";
047     } else if (evt.messageType == FlashGameCommunicator.ACTION_MESSAGE_LOBBY) {
048         taMessages.text += "Ich: "+inpMessage.text+"\n";
049         inpMessage.text="";
050         inpMessage.enabled=true;
051         btnSendMessage.enabled=true;
052     } else if (evt.messageType == FlashGameCommunicator.ACTION_MESSAGE_ROOM) {
053         taMessages.text += "Ich (Raum): "+inpMessage.text+"\n";
054         inpMessage.text="";
055         inpMessage.enabled=true;
056         btnSendMessage.enabled=true;
057     } else if (evt.messageType== FlashGameCommunicator.ACTION_ROOM_CREATE) {
058         taMessages.text += "Der Raum wurde erfolgreich erstellt.\n";
059         comm.roomList();
060         cbRoomLocked.enabled=true;
061         checkEnabled();
062     } else if (evt.messageType==FlashGameCommunicator.ACTION_ROOM_LIST) {
063         taMessages.text+="Raumliste empfangen (" +evt.dataArray.length + " Räume)\n";
064         listRooms.removeAll();
065         for (var i=0; i<evt.dataArray.length; i++) {
066             listRooms.addItem({label:evt.dataArray[i].roomName, data:evt.dataArray[i].room
067         }
068     } else if (evt.messageType==FlashGameCommunicator.ACTION_ROOM_LEAVE) {
069         checkEnabled();
```

```
070     comm.roomList();
071 } else if (evt.messageType == FlashGameCommunicator.ACTION_ROOM_JOIN) {
072     taMessages.text += "Ein Raum wurde betreten.\n";
073     checkEnabled();
074     comm.roomList();
075 } else if (evt.messageType == FlashGameCommunicator.ACTION_USER_LIST) {
076     taMessages.text += "Benutzerliste:\n";
077     for (var i=0; i<evt.dataArray.length; i++) {
078         taMessages.text += (i+1)+". "+evt.dataArray[i].userName+" (" +evt.dataArray[i].
079     }
080 } else if (evt.messageType == FlashGameCommunicator.ACTION_USER_LIST_ROOM) {
081     taMessages.text += "Benutzerliste des Raumes:\n";
082     for (var i=0; i<evt.dataArray.length; i++) {
083         taMessages.text += (i+1)+". "+evt.dataArray[i].userName+" (" +evt.dataArray[i].
084         if (i==0) {
085             taMessages.text += "MASTER\n";
086         } else {
087             taMessages.text += "\n";
088         }
089     }
090 } else if (evt.messageType == FlashGameCommunicator.ACTION_ROOM_INFORMATION) {
091     taMessages.text += "roomName: "+evt.roomName+", capacity: "+evt.roomCapacity+", ro
092 }
093 }
094 function failed(evt:GameEvent) {
095     if (evt.messageType == FlashGameCommunicator.ACTION_ROOM_CREATE) {
096         taMessages.text += "Der Raum konnte nicht erstellt werden.\n";
097         checkEnabled();
098     } else if (evt.messageType == FlashGameCommunicator.ACTION_ROOM_JOIN) {
099         taMessages.text += "Der Raum konnte nicht betreten werden.\n";
100     } else {
101         taMessages.text += "Aktion fehlgeschlagen.\n";
102     }
103 }
104 function kicked(evt:GameEvent) {
105 }
106 function sendNameHandler(evt:MouseEvent):void {
107     if (comm.isConnected()) {
108         comm.sendMessage(inpName.text);
109     }
110 }
111 function sendMessageHandler(evt:MouseEvent):void {
112     if (inpMessage.text.length<=0) {
113         return;
114     }
115     if (!comm.isConnected()) {
116         return;
117     }
118     if (comm.isInRoom()) {
119         comm.sendMessageRoom(inpMessage.text);
120     } else {
121         comm.sendMessageLobby(inpMessage.text);
122     }
123     inpMessage.enabled=false;
124     btnSendMessage.enabled=false;
125 }
126
127 function checkEnabled():void {
128     if (comm.isConnected()) {
129         btnSendName.enabled=true;
130         btnSendMessage.enabled=true;
131         btnRefreshRooms.enabled=true;
132         btnUserList.enabled=true;
133         btnDisconnect.enabled=true;
134         if (comm.isInRoom()) {
135             btnCreateRoom.enabled=false;
136             nsCapacity.enabled=false;
137             btnJoinRoom.enabled=false;
138             btnLeaveRoom.enabled=true;
```

```
139         btnUserListRoom.enabled=true;
140         btnRoomInformation.enabled=true;
141     } else {
142         btnCreateRoom.enabled=true;
143         nsCapacity.enabled=true;
144         btnLeaveRoom.enabled=false;
145         btnUserListRoom.enabled=false;
146         btnRoomInformation.enabled=false;
147         cbRoomLocked.enabled=false;
148     }
149 } else {
150     btnSendName.enabled=false;
151     btnRefreshRooms.enabled=false;
152     btnCreateRoom.enabled=false;
153     btnUserList.enabled=false;
154     nsCapacity.enabled=false;
155     btnSendMessage.enabled=false;
156     btnDisconnect.enabled=false;
157     btnJoinRoom.enabled=false;
158     btnLeaveRoom.enabled=false;
159     btnUserListRoom.enabled=false;
160     btnRoomInformation.enabled=false;
161     cbRoomLocked.enabled=false;
162 }
163 }
164
165 function createRoomHandler(evt:MouseEvent):void {
166     if (!comm.isConnected()) {
167         return;
168     }
169     if (comm.isInRoom()) {
170         return;
171     }
172     if (inpRoomName.text.length<=0) {
173         return;
174     }
175     comm.createRoom(inpRoomName.text, nsCapacity.value);
176 }
177
178 function disconnectHandler(evt:MouseEvent):void {
179     if (!comm.isConnected()) {
180         return;
181     }
182     comm.disconnect();
183 }
184
185 function refreshRoomsHandler(evt:MouseEvent):void {
186     if (!comm.isConnected()) {
187         return;
188     }
189     comm.roomList();
190 }
191
192
193 function leaveRoomHandler(evt:MouseEvent):void {
194     if (!comm.isConnected()) {
195         return;
196     }
197     if (!comm.isInRoom()) {
198         return;
199     }
200     comm.leaveRoom();
201 }
202
203 function listRoomsHandler(evt:MouseEvent):void {
204     if (listRooms.selectedItem==null || comm.isInRoom()) {
205         btnJoinRoom.enabled=false;
206     } else {
207         btnJoinRoom.enabled=true;
```

```
208     }
209 }
210
211 function joinRoomHandler(evt:MouseEvent):void {
212     if (!comm.isConnected()) {
213         return;
214     }
215     if (comm.isInRoom()) {
216         return;
217     }
218     var si:Object = listRooms.selectedItem;
219     if (si==null) {
220         return;
221     }
222     comm.joinRoom(si.data);
223 }
224
225 function userListHandler(evt:MouseEvent):void {
226     if (!comm.isConnected()) {
227         return;
228     }
229     comm.userList();
230 }
231
232 function userListRoomHandler(evt:MouseEvent):void {
233     if (!comm.isConnected()) {
234         return;
235     }
236     if (!comm.isInRoom()) {
237         return;
238     }
239     comm.userListRoom();
240 }
241
242 function roomInformationHandler(evt:MouseEvent):void {
243     if (!comm.isConnected()) return;
244     if (!comm.isInRoom()) return;
245     comm.roomInformation(comm.getRoomId());
246 }
247
248
249
```