Use Case 3.3: Chat

Author: Yasin Garip

Characteristic Information

Goal in Context: The user can communicate over the chat with the other client players.

Scope: Network

Level: Subfunction

Primary Actor: User

Stakeholders & Interests:

- User: Wants to send and receive chat messages to and from remote human players

Preconditions: Lobby is created and a message in the CHAT-GUI is sent.

Success Guarantee (Postconditions): Message is sent to all other player clients and displayed on the chat.

Trigger: Type in chat and click button send message

Main Success Scenario

1. User types in Chat/TextArea a message.
2. User clicks on the button “send message”.
3. The message gets displayed on every client connected to the server
4. TextBox gets cleared.

Extensions

*a. At any time, System fails:
1. System reopens system, logs in, and requests recovery of prior state.
2. System reconstructs prior state.
   2a. System detects anomalies preventing recovery: System signals error and records the error. System starts new tutorial.
3. Warning sign that the game crashed appears.

1. User types nothing in Textarea:
1. Send Button is grey and not activated to send messages.
3. Server didn’t get the text message:
1. TextBox does not get cleared
2. Error Message is shown

Sub-Variations

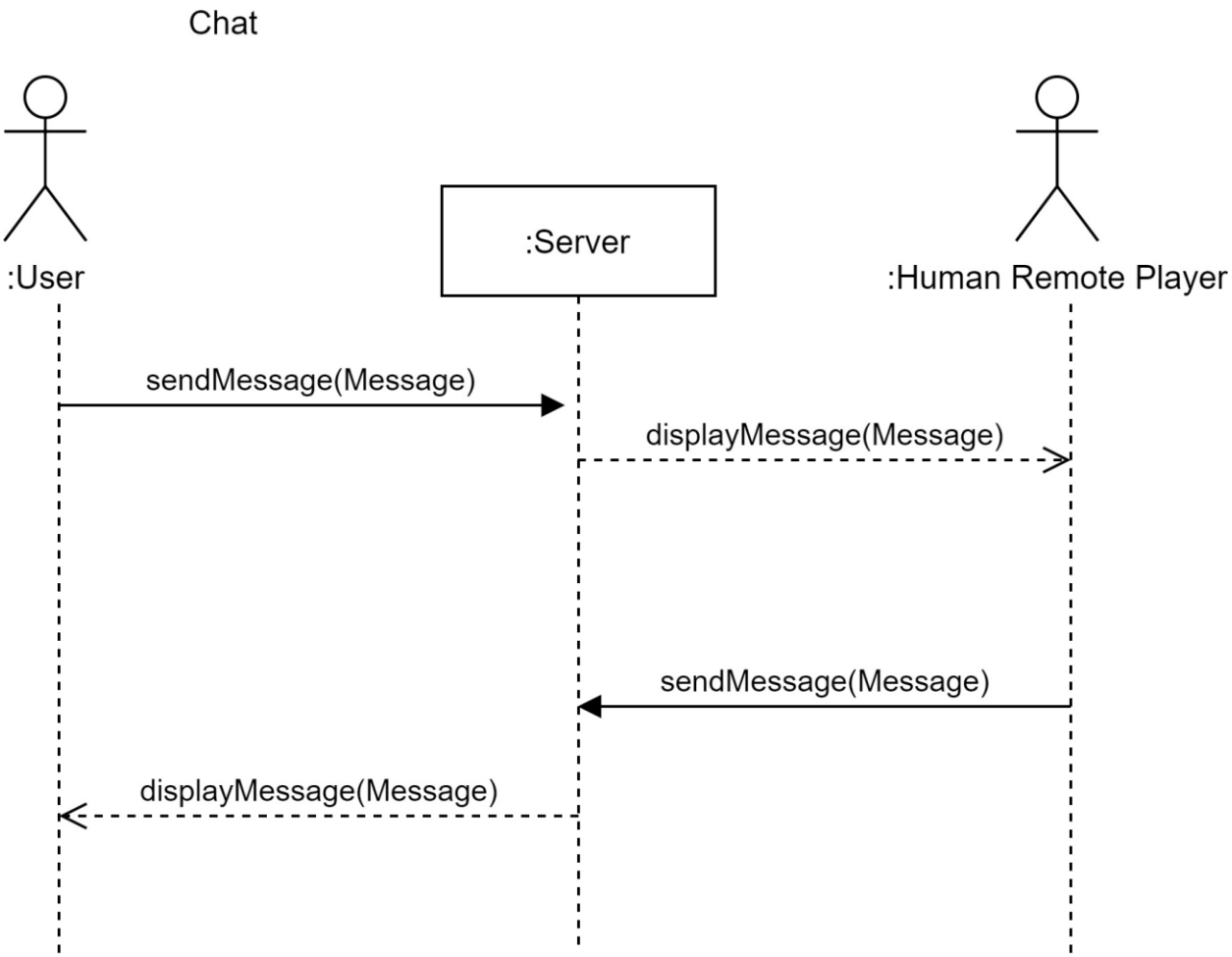
2. The User can press the “enter key” button to send the message when the textbox is clicked.

Due Date

10.05.2021

System Sequence Diagram

Author: Yasin Garip



Operation Contracts

Author: Yasin Garip

Contract 3.3.1: sendMessage

Operation: sendMessage(message : String) : void

Cross References: UC 3.3: Chat

Preconditions:

- User typed in a message in the textArea (GUI) and clicked on 'Send'

Postconditions:

- Text message was associated with Server s at s.chatLog