

Pflichtenheft und technische Spezifikationen

Projektname: Kooperatives Design

Mitarbeiter: Jannis Camille Brüns, Jerrey Glen Mekountchou Talla

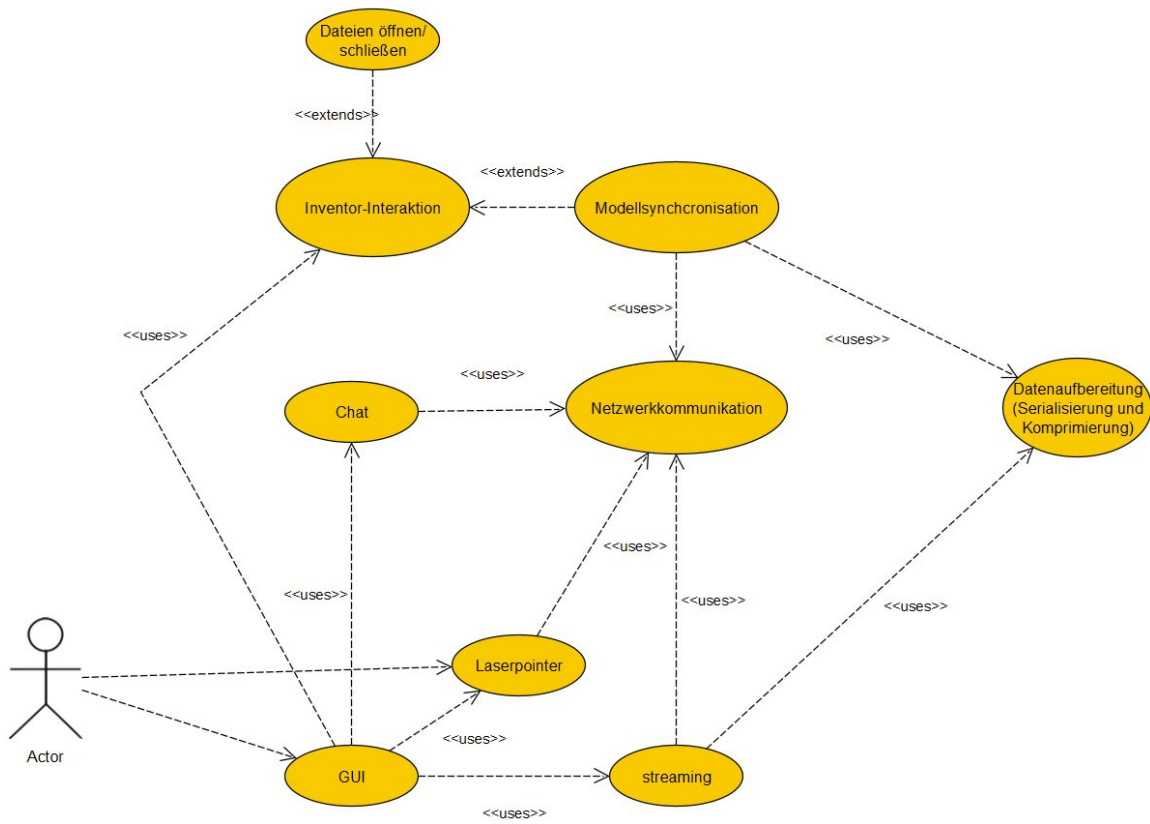
Inhaltsverzeichnis

Thema	Seite
FeatureListe	3
Use-Cases	4
Aktivitätendiagramme	5
Komponentendiagramm und Aufteilung	9
Gui-Mockup	10
Schnittstellen	11

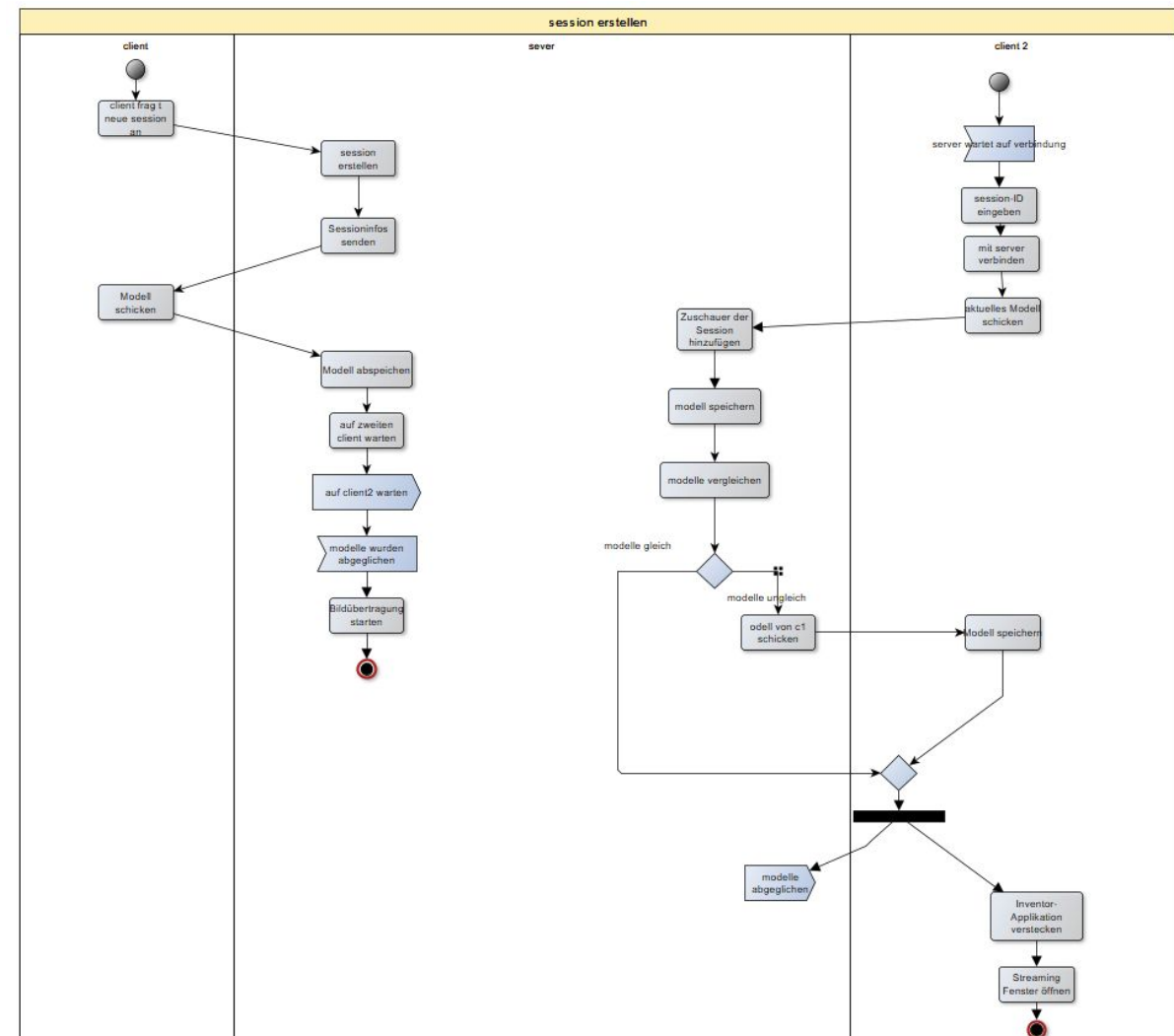
1. Feature-Liste

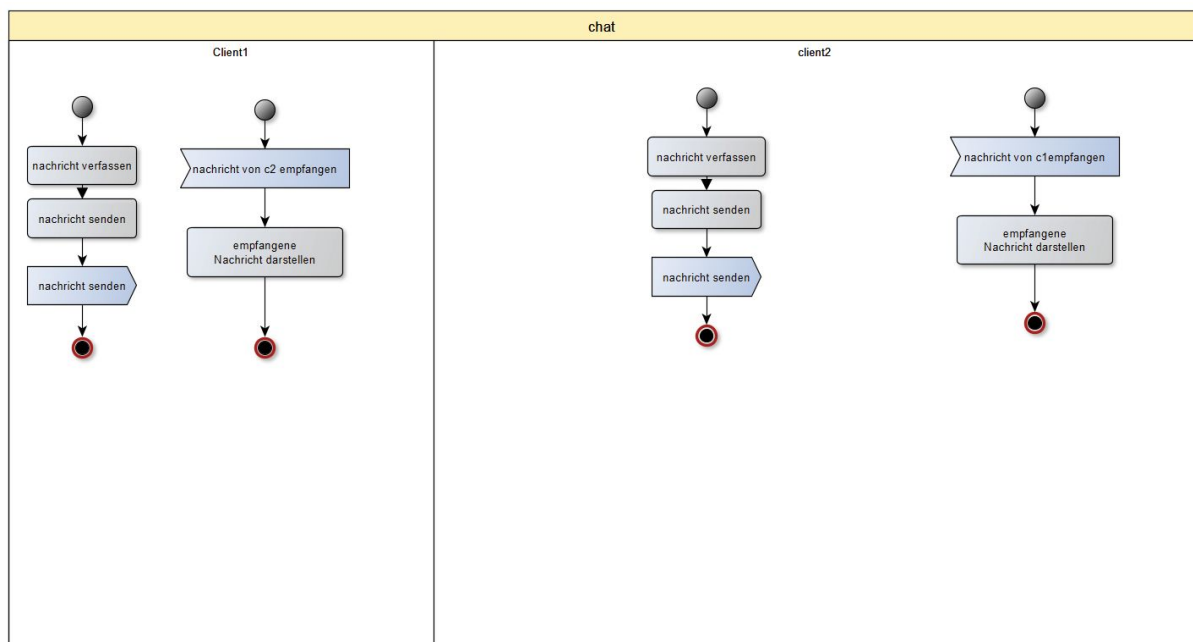
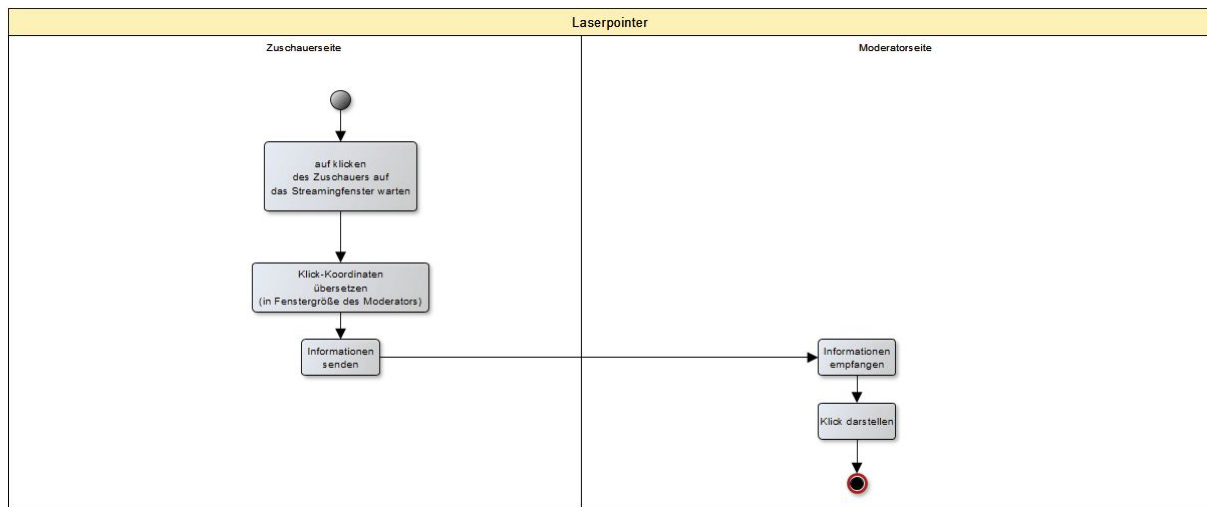
Feature-ID	Priorität	Beschreibung	Aufwand
Kommunikation mit Inventor	Sehr hoch	Dateien öffnen und schließen. Fenster anzeigen und verstecken. Knöpfe darstellen	mittel
Daten-aufbereitung	Sehr hoch	“Streambilder”, Chat-Nachrichten, und session informationen so verpacken, dass sie fehlerfrei wieder entpackt und interpretiert werden können	mittel
Netzwerk-kommunikation	Sehr hoch	Netzwerkkommunikation zwischen Rechner und Server. Bereitstellung verschiedener Funktionen um Zeichenketten, zahlen und Dateien hin und her zu schicken und zu empfangen.	hoch
GUI	Hoch	Einfach gehaltene Benutzeroberfläche, mit Knöpfen, um die Session zu starten, einer Session beizutreten, die Session zu verlassen oder sie zu beenden, das Chat Fenster zu öffnen und einer Anzeige für Informationen über die laufende Session. Im Chat werden die Nachrichten beider Beteiligten mit Zeitstempel und Absender dargestellt und die Eingabe von Nachrichten ermöglicht. Das Chatfenster kann geschlossen und wieder geöffnet werden. wenn man Zuschauer ist wird Inventor unsichtbar und ein andere Fenster soll sich öffnen, welches den Inhalt der Inventor-Applikation des Moderators zeigt. Dieses Fenster soll wiederum auch Knöpfe enthalten um die Session zu steuern.	hoch
“Laserpointer”	Mittel	ermöglicht es dem Zuschauer irgendwo zu klicken und die Position des Klicks dem Moderator mitzuteilen, um so auf Dinge hinweisen zu können.	mittel
Session-Verwaltung	Mittel	Erstellen mit Meeting-Id, Beitreten und Beenden von Sessions mit Rollenverwaltung	mittel
Streaming	sehr Hoch	Übermitteln des Inhalts der Inventor-Applikation des Moderators an den Zuschauer in einem bestimmten Intervall.	mittel

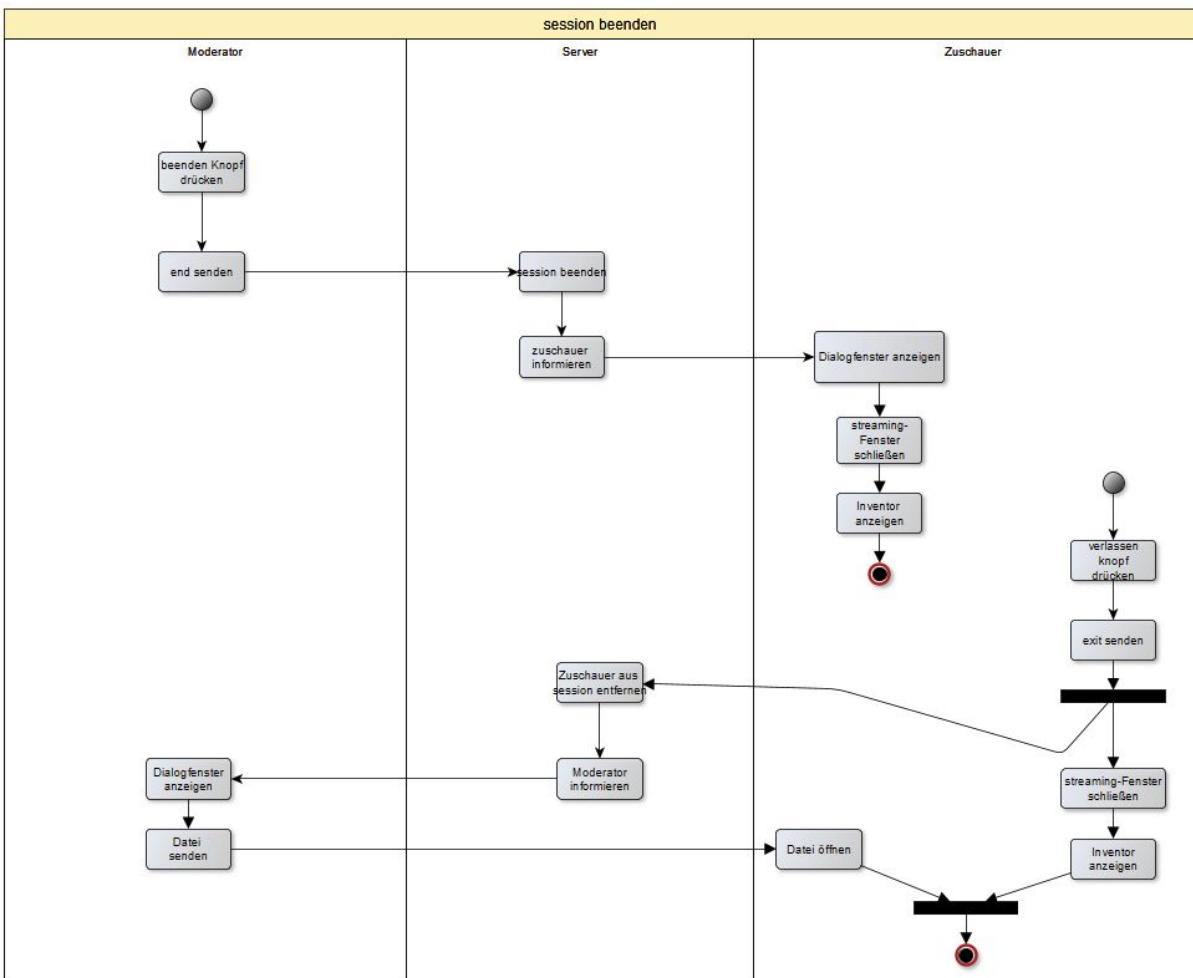
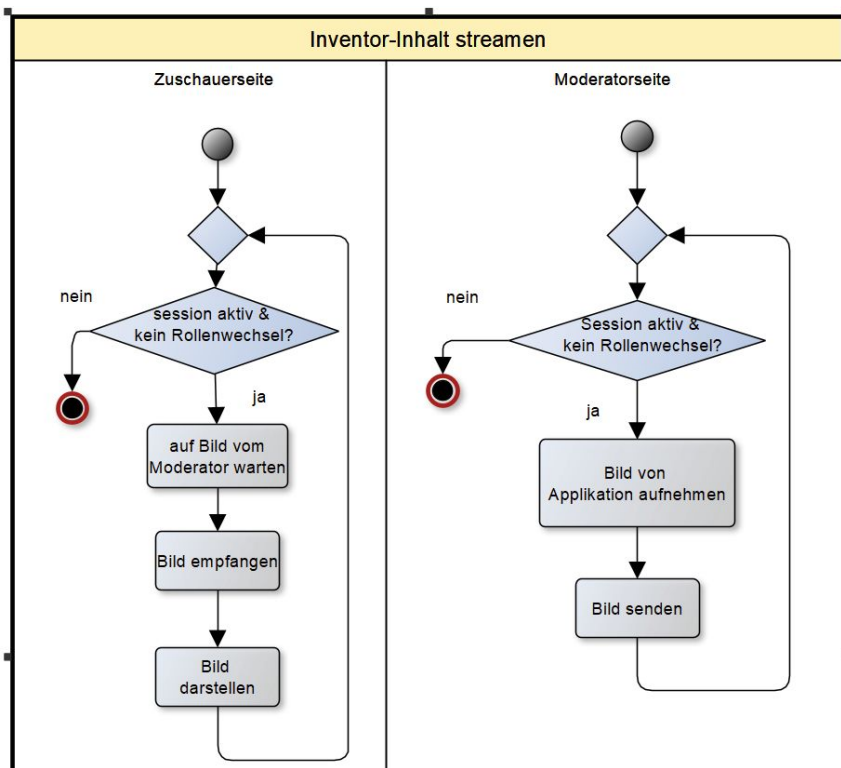
Use-Cases

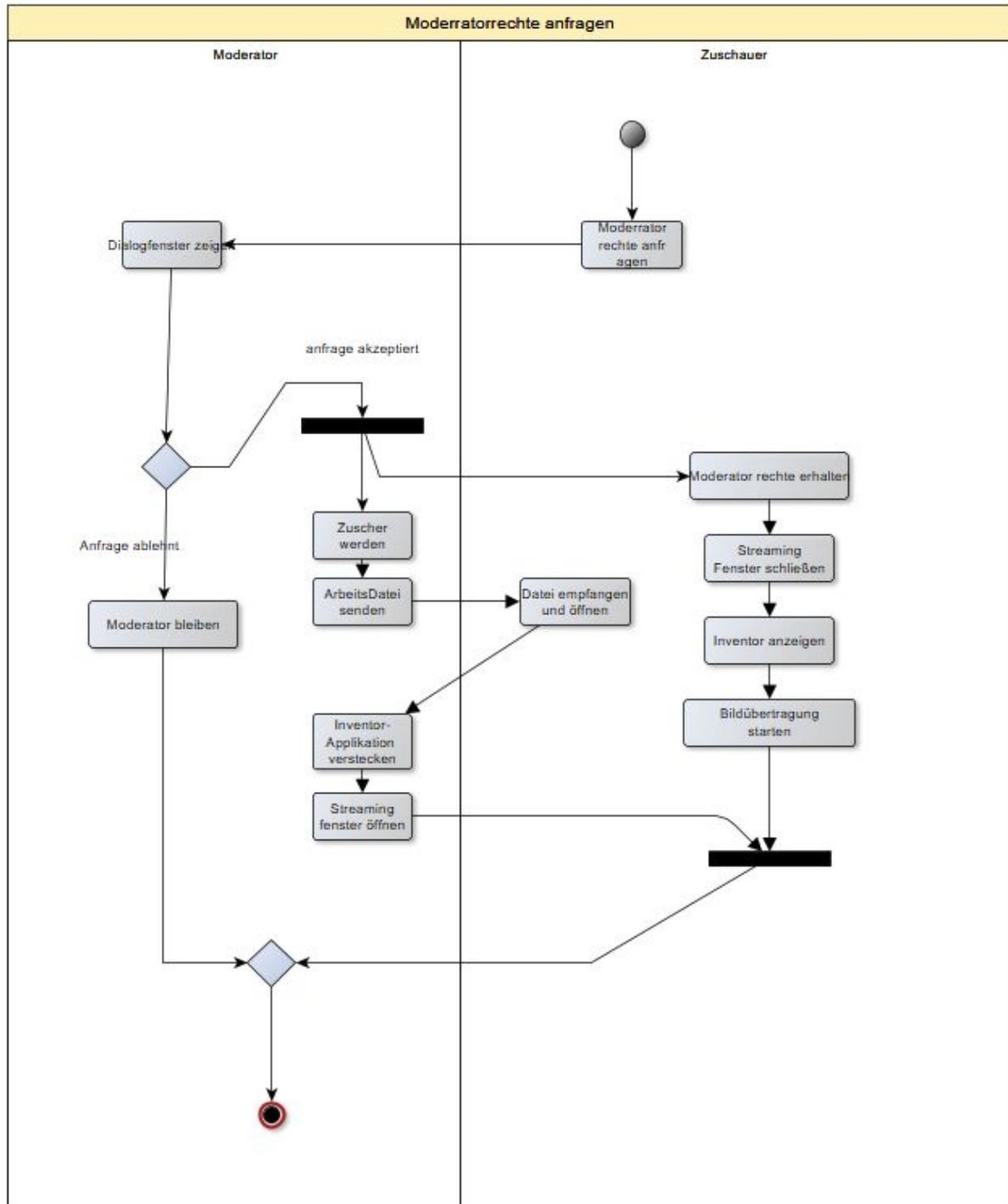


Aktivitäten-diagramme

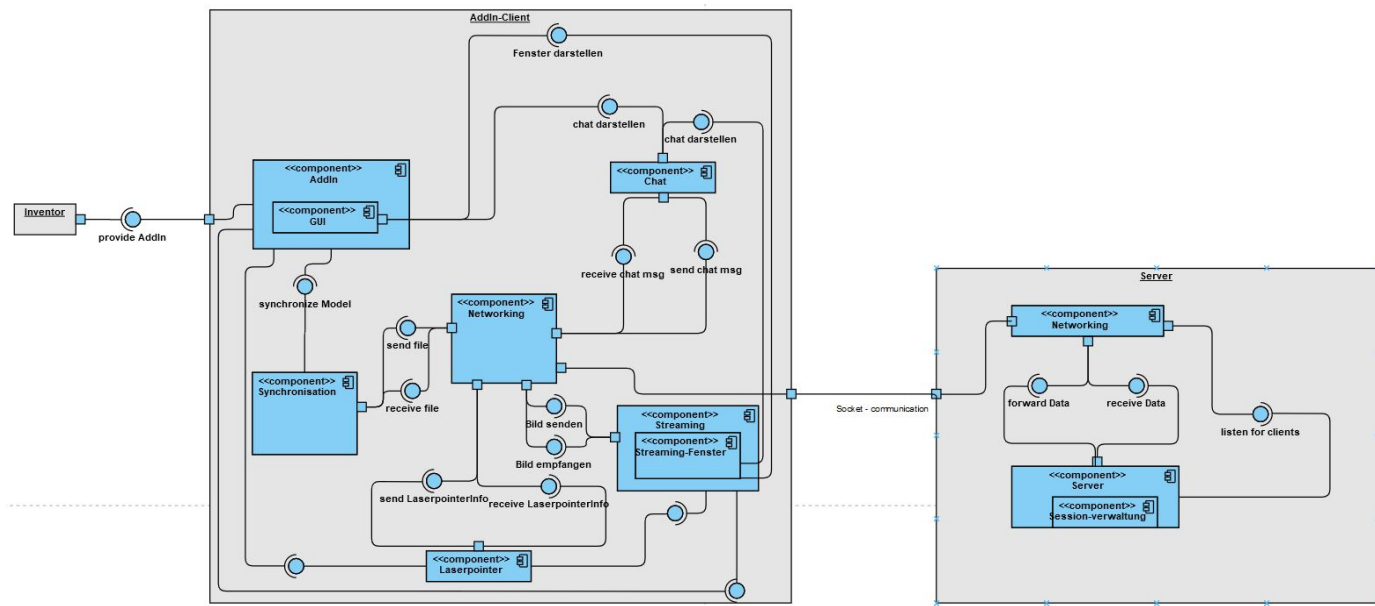








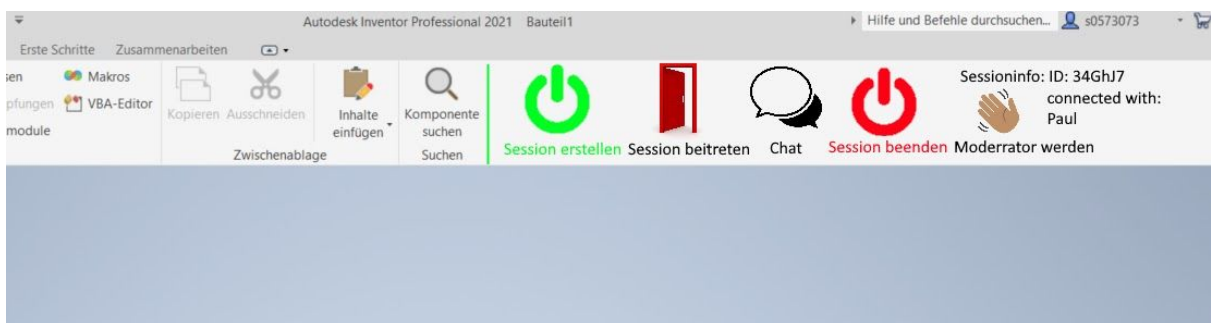
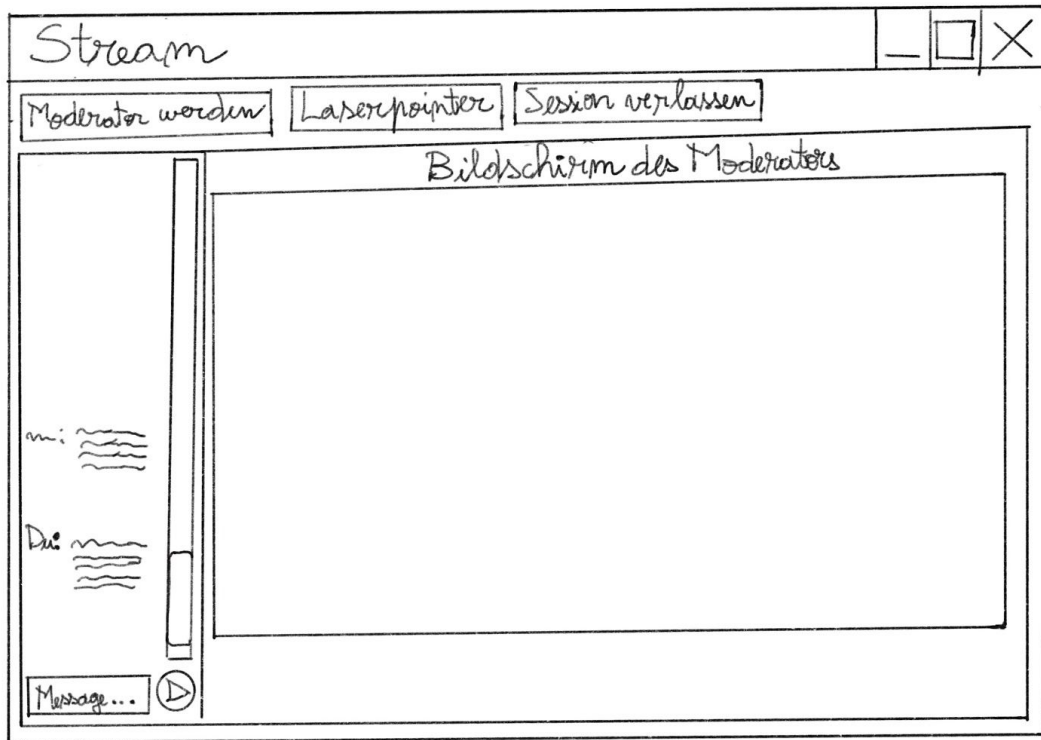
Komponentendiagramm



Aufteilung

Jannis Camille Brüns:	Server-system(Networking, Server), Networking for AddIn-Client, Laserpointer
Jerrey Glen Mekountchou Talla	AddIn, Synchronisation, Streaming, Chat

GUI-Mockup



Interne Schnittstellen

Komponente	Schnittstelle
AddIn	Event: documentchanged Implementierung von Inventor.ApplicationAddInServer
Chat	Void: newMSGCallBack() Void: display() Void: hide()
GUI	
Synchronisation	void: synchronize(byte[] FileData) Void: SynchronizeCallBack(byte[] FileData)
Networking client side	Event: chatMSGAvailable Void: sendChatMSG(string msg) Event: IMGAvailable Void: sendIMG(byte[] ImageData) Event: laserPointerInfoAvailable Void: sendLaserPointerInfo(LaserPointerInfo lpinfo) void: connect(string sessionid) void:disconnect()
Laserpointer	static void: laserPointerCallback(LaserPointerInfo: lpinfo) static void: start() static void: stop()

Server	int port List<session> sessions void: start() void: stop() void: deleteSession(Session s) Server(int: port)
Networking server side	void: startAsyncListening() void: forward(byte[] Data)
Streaming	void displayStreamingWindow() void: hideStreamingWindow() void: IMGreceivedCallBack(byte[] ImageData) void: sendIMG(byte[] imgdata)
Session	List<client> clients string: SessionId void: Add(client) void: removeClient(client) void: delete()
SessionManager	List<Session>: sessions SessionManager: Instance void: delete(Session s) void: createnew()

externe Schnittstellen

System	Schnittstelle
Client System	Inventor.ApplicationAddInServer