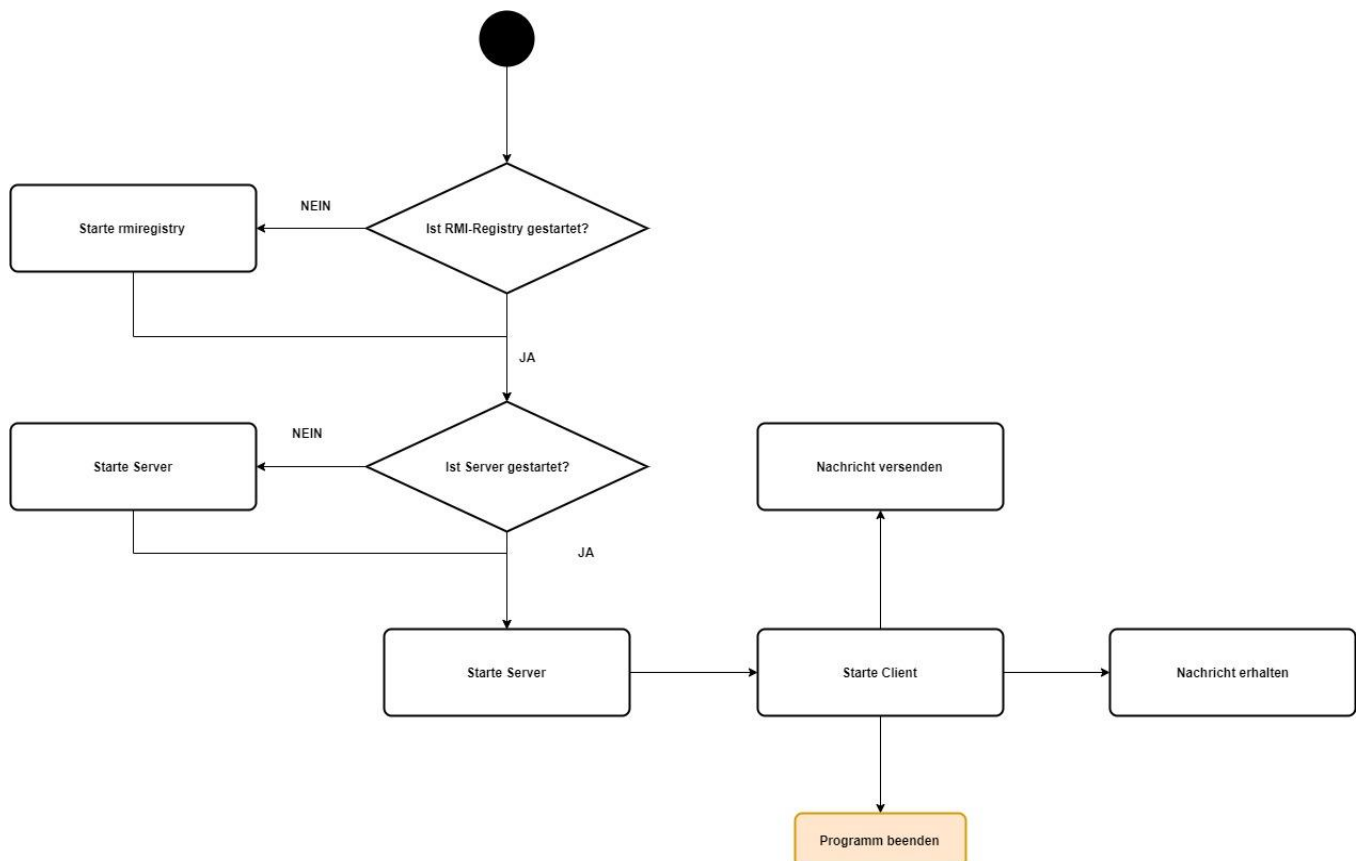


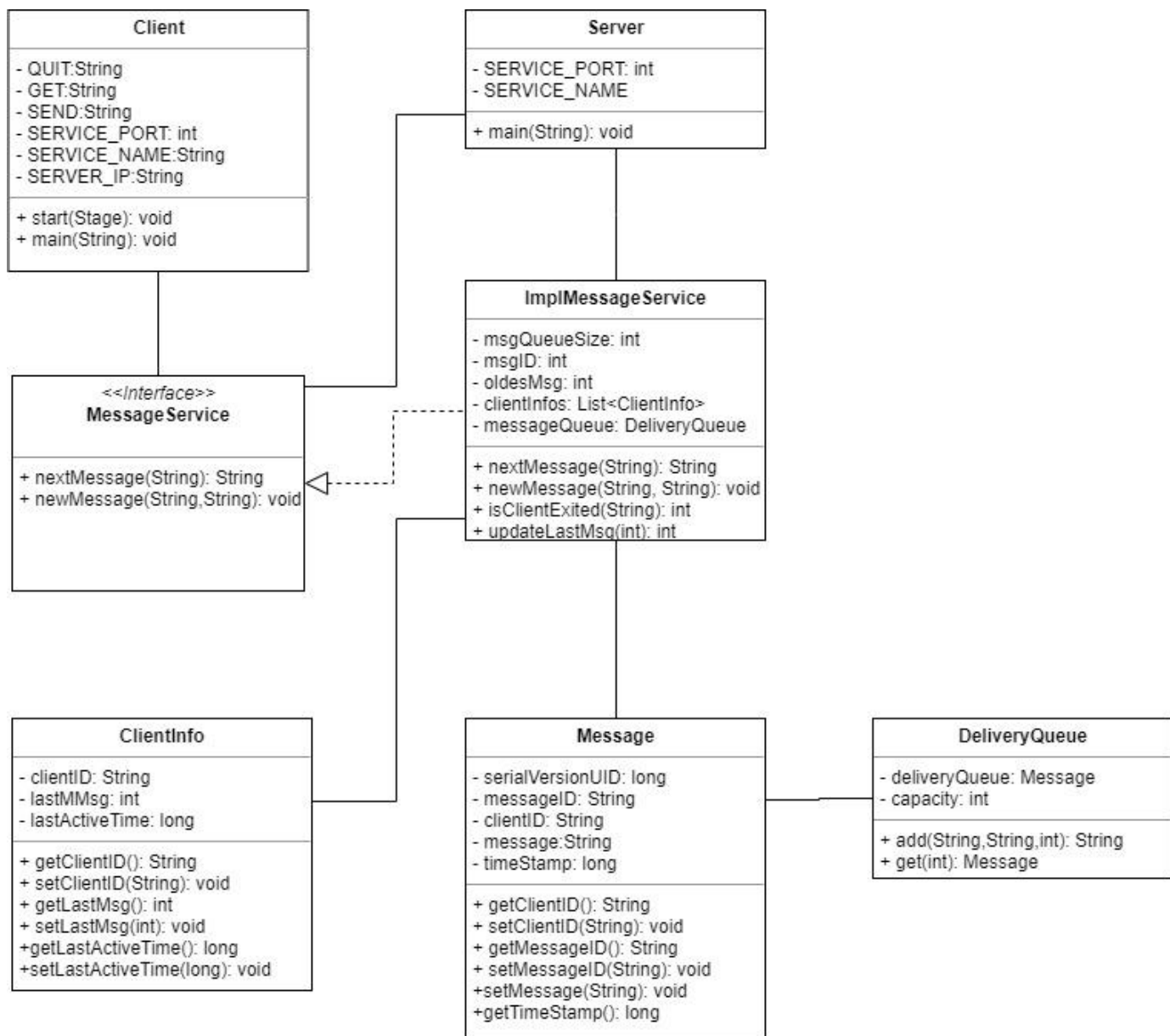
# Client/Server-Anwendung „Verteilte Nachrichten-Queue“ – Konzept

In dieser Aufgabe wird eine Java RMI-Schnittstelle genutzt, um Nachrichten zwischen mehreren Clients auszutauschen, welche mit einem Server verbunden sind. Die Remote Method Invocation (RMI) ist eine Art des Remote Procedure Call (RPC), welche in Java genutzt wird<sup>1</sup>. RPC realisiert die Interprozesskommunikation, welches den Informationsaustausch zwischen Prozessen ermöglicht. Daher können mit RMI Methoden entfernter Objekte aufgerufen werden. Der Server nutzt die RMI-Registry als Namensdienst, sodass der Client darüber auf die entfernten Methoden des Servers zugreifen kann.



1 - Allgemeiner Ablauf des Starts der Anwendung

Der Server versucht anfangs bei der RMI-Registry ein entferntes Objekt zu registrieren<sup>1</sup>, weshalb diese als erstes gestartet werden muss über den Aufruf „start rmiregistry“. Wenn der Server läuft kann der Client gestartet werden. Der Client hat, in Form eines „Stubs“, einen Stellvertreter für das entfernte Objekt beim RMI-Server<sup>2</sup> und kann auf die entfernten Methoden zugreifen. Der Server hat hingegen einen „Skeleton“, welcher der Stellvertreter für das aufrufende Objekt ist<sup>2</sup>. Der Client kann dann über die Rückgabewerte der Methoden die Nachrichten aus der DeliveryQueue erhalten und auch an die DeliveryQueue senden.



## 2 - Klassendiagramm

Die GUI wird im Client realisiert und auch beim Starten des Clients gestartet. Über die GUI wird dem Benutzer ermöglicht zu entscheiden, ob eine Nachricht abgerufen wird, versandt wird oder ob das Programm beendet werden soll. Bei dem Fehlerfall, dass der Client die Verbindung zum Server verliert, soll der Client nicht abstürzen, sondern die Exception fangen und weiterhin versuchen den Server zu erreichen und es wird eine entsprechende Meldung ausgegeben. Der Server implementiert die at-most-once Fehlersemantik, also wenn der Client den Server nicht erreichen kann und weiterhin Requests sendet werden diese in einer Liste gespeichert und die Duplikate werden gelöscht.

## Quellen:

1 – Wikipedia: Remote Method Invocation

[https://de.wikipedia.org/wiki/Remote\\_Method\\_Invocation](https://de.wikipedia.org/wiki/Remote_Method_Invocation)

2 - Middleware in Java Kapitel 6 Seite 5: Struktur einer RMI-Anwendung

<https://www.informatik.uni-marburg.de/~mathes/download/k6.pdf>