

## Anwendung "Tratsch" / Klassen für Botschaften zwischen Client und Server

Klasse	Bedeutung	Erläuterungen
ClientBotschaftLogin	Client beantragt Anmeldung	
ClientBotschaftLogout	Client beantragt Abmeldung	
ClientBotschaftSendenTextnachricht	Client beantragt Senden einer Textnachricht an einen bestimmten Empfänger	Attribute absender und empfaenger enthalten die jeweiligen Benutzernamen
ServerBotschaftAngemeldeteBenutzer	Server sendet Liste aller angemeldeten Benutzer	Attribut angemeldeteBenutzer enthält die Liste der Benutzernamen
ServerBotschaftLoginNOK	Server meldet, dass Anmeldung nicht erfolgreich war	Attribut meldung erläutert, warum die Anmeldung nicht erfolgreich war
ServerBotschaftLoginOK	Server meldet, dass Anmeldung erfolgreich war	
ServerBotschaftLogoutErzwungen	Server erzwingt Abmeldung des Clients	
ServerBotschaftLogoutOK	Server meldet, dass Abmeldung erfolgreich war	
ServerBotschaftSendenTextnachrichtNOK	Server meldet, dass Senden der Textnachricht nicht erfolgreich war	
ServerBotschaftTextnachricht	Server sendet Textnachricht von einem bestimmten Absender an einen bestimmten Empfänger	Attribute absender und empfaenger enthalten die jeweiligen Benutzernamen

### Umgang mit den Botschaftsklassen (Beispiel)

TratschGemeinsam.jar in den Classpath aufnehmen

```
Botschaft botschaft = (Botschaft) inputStream.readObject();
handleBotschaft(botschaft);

private void handleBotschaft(Botschaft pBotschaft)
{
    if (pBotschaft instanceof ServerBotschaftAngemeldeteBenutzer)
    {
        ServerBotschaftAngemeldeteBenutzer botschaft = (ServerBotschaftAngemeldeteBenutzer) pBotschaft;
        // Aufruf der Methoden des Objekts "botschaft" usw.
    }
}
```