1. Spieler 1 Geld setzen (setMoney())
   1. -> Geben
2. Spieler 2 Geld setzen
   1. -> Geben
3. je 2 Karten austeilen (Sp1 + Sp2 + Dealer)
4. 3x Check „hat er BlackJack 21“?
5. Spieler 1 „hit“ oder „stay“
   1. hit -> noch eine Karte
      1. ist er <21:
         1. Check „hat er BlackJack 21“?
      2. ist er >21: Spieler verloren + Warten
   2. stay -> Boolean „letzteRunde“ = true
6. Spieler 2 „hit“ oder „stay“
   1. hit -> noch eine Karte
      1. ist er <21:
         1. Check „hat er BlackJack 21“?
      2. ist er >21: Spieler verloren + Warten
   2. Stay -> Boolean „letzteRunde“ = true
7. Dealer „hit“ oder „stay“
   1. ist er <17: hit
      1. ist er <21:
         1. Check „hat er BlackJack 21“?
      2. ist er >21: Spielzug verloren -> Spieler 1 gewinnt() + Spieler 2 gewinnt()

**wenn „letzteRunde“ = false** 🡪 **Schritt 5**

1. Karten aufdecken()  
   pruefeGewinner() (in der Methode wir gleich getMoney() ausgeführt + Meldung)

|  |  |  |
| --- | --- | --- |
|  | Server | Client |
| Schritt 1 | Step1: Geldbetrag eingeben   * dos write Int   Step1 -> Step2 | Step1: wartet auf   * dis readInt * Step1 -> Step2 |
| Schritt 2 | Step2: wartet auf   * dis readInt * Step2 -> Step3 | Step2: Geldbetrag eingeben   * dos write Int * Step2 -> Step3 |
| Schritt 3 | Step3: Nacheinander je Karte ziehen und übermitteln  writeUTF und DataOStream  (macht das 6 mal)  Speichert Karten von allen | Empfängt nacheinander die Daten  Nur readUTF und DataInputStream |
| Schritt 4 | Prüft ob gewonnen oder nicht  übermitteln  writeUTF und DataOStream | Nur readUTF und DataInputStream |
| Schritt 5 | Karte ziehen und übermitteln  writeUTF und DataOStream  Wenn stay: „stay“ | Nur readUTF und DataInputStream  Wenn „stay“ ankommt mach nix |
|  |  |  |