

# Meilensteinpräsentation V

Programmierprojekt 2020

Implementierung von **Skip-Bo** ©

Gruppe: Skip-Bros



Manuela Wildi



Rao Rohan Girish



Guillaume Joyet



Janni Batsilas



Quelle: [mercadolibre.com](https://www.mercadolibre.com)



Skip-Bro: Regeln, Ziel, Mechanik



Spieldemonstration



Technische Eigenheiten



Lektion gelernt



**Um mitzuspielen:**

**IP Adresse: 10.34.58.178:8015**

**Port: 8015**

**Link zum Download des JARs im Chat**



**Skip-Bro: Regeln, Ziel und Mechanik**



Spieldemonstration



Technische Eigenheiten




Lektion gelernt



# Skip-Bro: Regeln, Ziel und Mechanik

Tutorial



Manual Change name  
Info Games list  
Player list Leave game  
Ready Start Game

Chat High score  

[Info] This game was developed by Manuela, Janni, Guillaume and Rohan for cs-108 in 2020!  
[Info] Welcome to Skip-Bo, Rohan!  
[Info] janni joined the room. Say hi!  
[Info] janni is READY.  
[Info] Status changed to READY.  
[Info] Game is starting...  
[Info] It's your turn!  
[Info] It's janni's turn!  
[Info] It's Manuela's turn!  
[Info] It's your turn!  
[Info] It's janni's turn!  
[Info] It's Manuela's turn!  
[Info] It's your turn!  
[Info] It's janni's turn!  
[Info] It's Manuela's turn!  
[Info] It's your turn!  
[Info] It's janni's turn!

Chat with: lobby

Draw pile

Build piles

Manuela

janni

Your stock pile

Your Discard Piles

Your hand cards

2 cards left



Skip-Bro: Regeln, Ziel, Mechanik



**Spieldemonstration**



Technische Eigenheiten



Lektion gelernt



Skip-Bro: Regeln, Ziel, Mechanik



Spieldemonstration



**Technische Eigenheiten**



Lektion gelernt



- **Apache Log4j2**
- **MP3SPI**
  - Ermöglicht abspielen von mp3 Dateien
  - Intuitive Implementation
  - Eigenhändige Ergänzungen
- **WrapEditorKit Klasse**
  - Hilft zur korrekten Darstellung des Chats
- **Timer**
  - Verzögerter Ausführung von Tasks





Skip-Bro: Regeln, Ziel, Mechanik



Spieldemonstration



Technische Eigenheiten

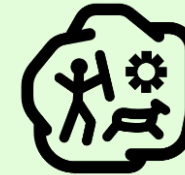
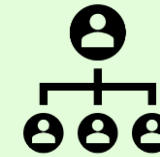


**Lektion gelernt**



# Lektion gelernt

- **Organisatorisch:**
  - Mehr Zeit für Neues einplanen
  - Code der Anderen laufend überfliegen
  - Bessere Strukturierung von Beginn an
- **Code vom Spiel:**
  - Mehr Vererbung
  - Wo möglich Enum-Typen
  - Strukturierte Vergabe von Variablennamen
  - Verwendung von JavaFX
  - Testen auch übers JAR
- **JUnit-Tests:**
  - Zu Beginn Überlegungen zu den Tests
  - Genauer informieren
  - Vererbung und Interfaces erhöhen Testbarkeit





***Vielen Dank für eure Aufmerksamkeit und  
viel Erfolg bei den Prüfungen !***

Quelle: mercadolibre.com